Wearable Inertial Sensors for Motion Analysis in Respiration, Diet Monitoring, and Vehicular Safety Applications

A THESIS SUBMITTED TO THE FACULTY OF THE UNIVERSITY OF MINNESOTA BY

Gregory W. Johnson

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

Advisor: Rajesh Rajamani

July 2021

© Gregory W. Johnson 2021

All Rights Reserved

Acknowledgements

I would like to acknowledge first and foremost my advisor, Professor Rajesh Rajamani, for the opportunity and support to pursue research in his laboratory and the technical, professional, and personal mentorship he provided along the way. I will be forever grateful for his time and patience.

I would also like to acknowledge the mentorship of Dr. Paul (Paolo) Pianosi and the wonderful collaboration and enlightening education in respiratory analysis he enabled.

I would like to sincerely express my gratitude to the members of my PhD committee, Professor Max Donath, Professor Demoz Gebre-Egziabher, and Professor Zongxuan Sun, who were more than willing to spend time out of their busy schedules to review my work and ask pointed but always fair questions.

To past and present LISEC members Dr. Mahdi Ahmadi, Dr. Corey Cruttenden, Dr. Woongsun Jeon, Ali Nouriani, Navaneeth Pushpalayam, Zhenming Xie, Dr. Song Zhang, and Dr. Ye Zhang, thank you for the comradery and the endless interesting conversations on the way to coffee. I would especially like to thank Dr. Yan Wang for introducing me to IMUs and Direction Cosine Matrix estimation, along with Dr. Ryan Madson, Hamidreza Movahedi, and Dr. Heng Wang for the encouragement, expertise in estimation, and the kindness to make it through my presentations without falling asleep.

To my siblings Neil, Ellen, and Keith, and to my parents Steve and Gloria, thank you for the unconditional love and support. To my son Charlie and daughter Maggie, thank you for reminding me that the best things in life have nothing to do with work. To Ben, I am grateful for our enduring friendship now entering its third decade and for your humor when I really needed it.

Going back to graduate school full time for a PhD after working in industry for ten years was a dream long in the making, and it absolutely would not have not been possible without the encouragement, positive attitude, moral and financial support, dedication, and personal and professional sacrifices made by my wife, Sarah. Without the leverage of her own experience, wisdom, perspective, and love, I surely would not have seen this thing through to the end.

Dedication

To my family, for the endless encouragement and support.

Abstract

This thesis is concerned with the development and application of motion analysis algorithms based on signals from inertial measurement units (IMUs). In particular, the application areas discussed in the thesis are respiratory monitoring, dietary monitoring, and vehicular safety. Usage of IMUs for attitude and heading estimation has a rich legacy, but it is only in recent years that they have become low-cost commodity sensors found in nearly every smart phone and smart watch, making them particularly applicable sensors for everyday applications.

Despite the existence of well-established orientation estimation techniques, motion analysis using inexpensive wearable sensor applications targeted to the general population requires special attention. All three application areas discussed in this thesis require a similar approach to the estimation of motion variables in that they depend on the partial or full orientation of the device relative to the human user and/or the user's orientation relative to earth. However, the class of mobile-phone grade IMUs utilized here offer notoriously poor accuracy compared to much more expensive aerospace-grade IMUs. Inexpensive IMUs typically suffer from bias instability, which requires careful calibration or specialized algorithms. Further, full orientation estimation traditionally relies on the IMU's magnetometer to sense the geomagnetic field. But, the geomagnetic field is relatively weak and can often be dwarfed by magnetic fields from ferromagnetic objects routinely encountered in indoor environments. Thus, applications targeted for use by the general population must utilize algorithms that can overcome these limitations in a robust manner.

The first application area addressed is respiratory monitoring. The physical motions of the thoracoabdominal wall during respiration are important in many diseases, and differentiation of normal from abnormal respiratory kinematics can be used to monitor disease state. In this application, a novel wearable device is developed that allows for long-term, out-of-clinic monitoring and analysis of respiration under the assumption of static body position (non-ambulatory). In particular, the device measures respiratory accelerations at multiple points on the thoracoabdominal surface and estimates respiratory displacements along with a variety of clinically useful metrics. After careful removal of gravity from the acceleration measurement using a multiplicative Kalman Smoother, the algorithm double integrates and high pass filters the residual signal to obtain three-dimensional respiratory displacements. The accuracy is on the order of the accuracy of a reference optical motion tracking system, and this thesis presents an analysis of the factors contributing to displacement errors. From the displacements, a variety of

additional temporal, phasic, and volumetric respiratory variables may be estimated. After developing methods and discussing experimental results from a single subject for estimating respiratory displacements and subsequently several respiratory variables from these displacements, we then present the results from an initial small cohort IRB-approved study using the device. In the study, subjects wore the respiratory monitor while faced with a variety of airway occlusions. Despite the ultra-low respiratory rates encountered, the system was able to detect thoracoabdominal asynchrony with limited accuracy. Real-life medical situations involving respiratory distress are likely to present higher respiratory rates and thus higher potential for more accurate estimates. The developed system offers a combination of capabilities unmatched by existing technology in terms of its portability and the suite of respiratory variables it is able to estimate.

The second application area addressed in this thesis is the development of a novel Food Intake Monitoring (FIM) device. Typical methods for dietary tracking in obesity research such as the 24-hour food intake recall are well known to be inaccurate, and there is clear need for a device to automatically detect and capture eating events as an adjunct to these existing methods. In this thesis, a wrist-worn IMU and microcontroller are utilized to detect when a person is eating (under the assumption that the food is eaten primarily with the sensor-affixed arm), optimize the capture of the food being eaten using an on-board camera, and classify the obtained image as containing food or not. The detection, image capture, and classification modules are organized in a decision tree format, an approach which minimizes system power consumption while maximizing user privacy, as opposed to having a camera always on with constant wireless data being streamed. In the first iteration of the FIM, hand proximity to the mouth is decided based on two IMUs, one on the upper arm, and one on the lower arm. In the second iteration of the device, only a single IMU is utilized, and hand proximity is determined using the IMU's magnetometer along with a magnet worn on the body near the collar bone. Once hand-mouth proximity has been detected, it is shown that a simple linear Support Vector Machine is able to accurately classify eating activities versus other hand-near-mouth activities, such as teeth brushing and shaving. After eating is detected, the system takes an image of the food in front of the user using an on-board camera. The timing of the image capture is based on estimation of the device orientation relative to gravity using a straightforward Kalman Filter, and a method is developed that predicts optimal image capture timing using the gyroscope. Finally, it is shown that images may be classified as containing food or not using a special Convolutional Neural Network (CNN) adapted to microcontroller deployment using integer quantization.

The final health and safety application considered concerns vehicular safety and phone use while driving. Distracted driving due to phone or mobile device usage is one of the primary causes of vehicular accidents, and one approach to reducing such accidents is to automatically disable devices when the user is driving. In this thesis, IMU signals on a mobile phone or smart watch are utilized to determine whether or not the user is in the driver's seat of a moving vehicle, under the assumption that the device is in a static position inside the vehicle and close to level road grade. First, the algorithm must estimate the orientation of the device relative to the vehicle. As in the other applications, fundamental limitations of mobile-phone grade IMUs prevent estimation of orientation using traditional methods. Instead, the algorithm uses motion signals obtained during braking to determine the forward direction of the vehicle, while estimation of the gravity direction fully constrains the phone orientation. Once the orientation is determined, the pitch and roll dynamics encountered during braking and turning the vehicle are used to determine which quadrant of the vehicle the device is in relative to the vehicle's center of gravity. Successful identification of seat position is demonstrated first in simulation and then experimentally using data taken during real-world city driving conditions.

Table of Contents

List of Tables

List of Figures

Chapter 1:
Introduction and background material

Introduction

In recent decades, development and sales of wearable devices such as smart watches have increased greatly and have been widely adopted by the general population. As one example, the Apple Watch went on the market only in 2015, but as of December 2020 has 100 million units actively in use by some estimates [1]. One of the primary marketing points of these devices is the ability of always-on sensors to track, record, monitor, and even help guide decisions regarding our physical well-being. Simple applications like step counters offer a snapshot of one's daily physical activity, and a typical capability is to remind the user to meet a daily step target. While the use of step counting devices as a long-term intervention to increase daily physical activity has been shown to be remarkably effective [2], the suite of sensors common to smart devices can be applied to far more sophisticated applications in the broad category of human health and safety that have yet to be developed [3]. As one example of the opportunity that exists for large-scale automated health monitoring, researchers at Stanford completed a study in 2019 of over 400,000 Apple Watch users in an effort to use on-device optical sensors to detect atrial fibrillation [4]. A recent human safety application concerns fatigue monitoring in the workplace and using wearable sensors to guide decisions regarding work breaks and reassignment [3].

The goal of this thesis is to develop novel motion analysis systems and associated algorithms for three health and safety applications that are yet to be addressed by existing technology in the categories of respiratory monitoring, dietary monitoring, and vehicular safety. As will be shown, all of the applications involve a broadly similar approach of processing physical motion data from on-device sensors using novel algorithms and then making application-specific assessments, decisions, or classifications. One of the sensors commonly available on smart devices to measure physical motion is the Inertial Measurement Unit (IMU). IMUs come in many different configurations, but the typical 9-axis IMU package consists of a tri-axis accelerometer that measures gravity and linear accelerations, a tri-axis gyroscope that measures rates of rotation, and a tri-axis magnetometer that measures magnetic fields surrounding the sensor.

A classic application of IMUs is estimation of a rigid body's orientation relative to an inertial frame such as the earth. A rigid body has three degrees of rotational freedom. Under the

assumption of small linear acceleration, the accelerometer measurement of gravity fixes two of the three degrees of rotational freedom. Further, the body's rotation about the gravity vector can be fixed using the magnetometer measurement of the geomagnetic field. Together, the magnetometer and accelerometer essentially can provide a static or low bandwidth estimate of orientation. On the other hand, the gyroscope's measurements of body rotational rates can be integrated from a known initial state to also estimate orientation. But bias in the gyroscope results in integration drift, so the gyroscope alone is beneficial only across short time scales. Using a Kalman filter, observer, or other sensor fusion algorithm allows the magnetometer- and accelerometer-based orientation estimate to be combined with the gyroscope-based estimate to provide both a drift-free and a high-bandwidth orientation estimate [5].

Despite a rich history of estimation algorithms initially developed for aerospace applications [5], the class of inexpensive mobile-phone grade IMUs generally used in wearable devices suffer from several fundamental issues. As mentioned, integration drift is a fundamental issue not only for gyroscopes but also for accelerometers in the case of double integration to obtain displacement estimates, issues exacerbated in inexpensive IMUs due to bias instability and noise levels [6]. A second fundamental issue with using IMUs in wearable devices and indoor applications concerns reliance on the geomagnetic field to constrain the third degree of rotational freedom of the rigid body. The geomagnetic field is relatively weak and can easily be dwarfed by common objects encountered in our daily lives. Calibration of a tri-axis magnetometer usually relies on the assumption that the measured magnetic field is a fixed vector (the geomagnetic field) at a particular location in space, and so the response surface of the measured field as the tri-axis magnetometer rotates should be ideally spherical. In reality, the uncalibrated magnetometer response surface is an ellipsoid, with biases and scaling errors contributing to distortions of the response surface. Calibration then essentially is an ellipsoid fitting problem. But, given that the cleanliness of the magnetic environment cannot generally be guaranteed either during calibration or active use, relying on the magnetometer for accurate horizontal plane rotation estimates is problematic.

One of the initial targeted applications for this thesis was at-home physical therapy. In particular, we were interested in automatically monitoring the limits of neck rotation during home-based therapy using a head-worn IMU. Neck rotation measurement relies almost entirely on the magnetometer, since the rotation occurs primarily about the vertical axis when the patient is sitting or standing. Unfortunately, nearby ferrous objects, such as steel furniture, can easily impact the estimation accuracy. Even with careful and repeated magnetometer calibration, we

found that rotation error when compared to a laboratory-grade optical motion capture system was often upward of 10° due to disturbances of the geomagnetic field inside the building, as demonstrated in Fig. 1 using data taken from a neck rotation experiment. In fact, previous research found heading estimate differences nearing 180° inside one biomedical motion laboratory [7], and so an error of 10° might be relatively low in practice. This error is compounded in multi-limb tracking applications, such as hand tracking, when a single IMU is needed for each limb in the kinematic chain between the torso and the hand. These initial findings suggested to us that wearable sensors targeted to the general population should generally not rely on the geomagnetic field when the magnetic cleanliness of the surrounding environment cannot be controlled. One of the overarching themes of this thesis is to develop hardware and algorithms that overcome these fundamental limitations of mobile phone-grade IMUs for robust, wearable health and safety applications.

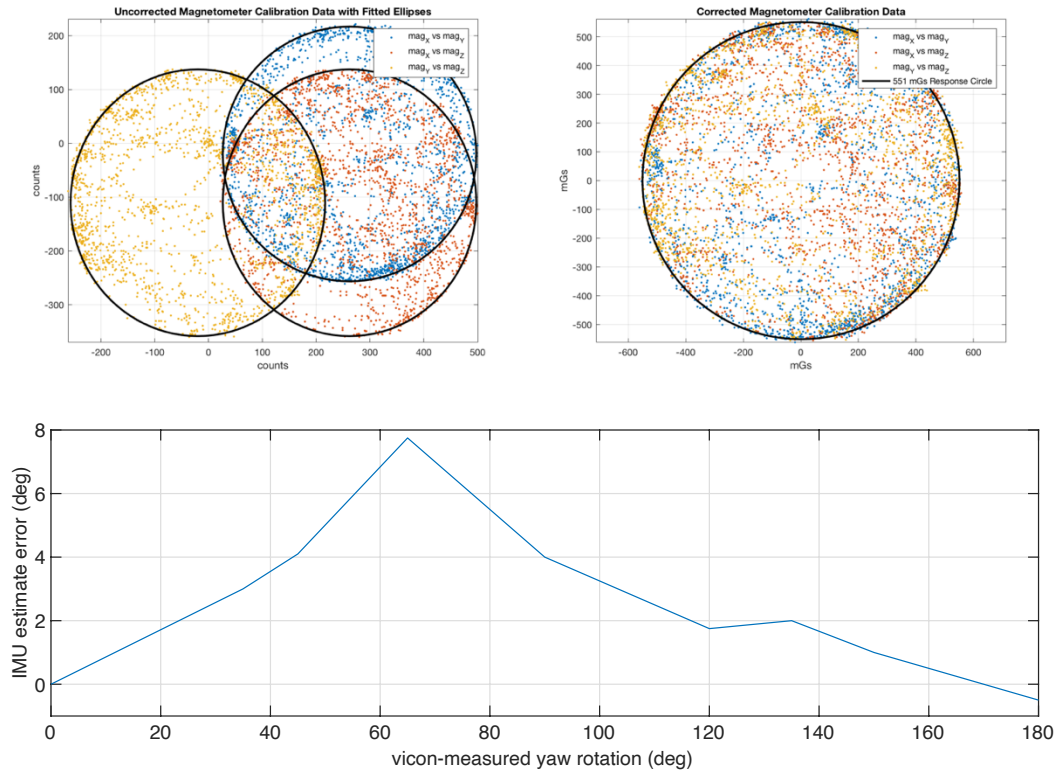

Fig. 1. Real-world magnetometer calibration
Magnetometer calibration (top left before and top right after) and heading accuracy (bottom) in an interior setting. Common household objects can easily dwarf the geomagnetic field, a fundamental limitation of using IMUs for orientation estimation in wearable sensors.

Thesis Contributions

This thesis is concerned with the development of wearable IMU-based systems for motion analysis in novel human health and safety applications. The main contributions of the thesis are:

1. Development of the hardware and associated algorithms for a novel wireless, wearable, and portable respiratory monitor that is shown experimentally to:
    - accurately estimate respiratory displacements under the assumption of non-ambulatory body motion at multiple points on the thoracoabdominal wall through careful removal of the gravity signal from acceleration measurements using a special multiplicative Kalman Smoother and double integration of the residual signal.
    - accurately estimate a suite of clinically-meaningful respiratory variables related to the temporal, phasic, and volumetric parameters of breathing.
    - detect thoracoabdominal asynchrony resulting from labored breathing due to an obstructed airway, as demonstrated in a limited IRB-approved study of five participants.
2. Development of the hardware and associated algorithms for a novel wrist-worn Food Intake Monitor. Using the assumption that eating primarily takes place using the dominant hand, we show experimental demonstration of the device's capability to:
    - detect proximity of the hand relative to the mouth in two ways: (1) using a multiple IMU approach and direction cosine matrix estimation and (2) a simple wrist-only setup along with a permanent magnet worn on the collar bone.
    - classify the subset of activities in which the hand is near the mouth as eating or non-eating using a support vector machine and features extracted from motion data obtained from the wrist-worn IMU.
    - automatically capture an image of the food in front of the user using a wrist worn camera and classify the captured image as containing food or not using a specially adapted on-board convolutional neural network.
3. Development of a novel algorithm for localization of a personal device within a moving vehicle (under the assumptions of typical city driving conditions and close to level grade) and experimental demonstration that the algorithm:

- estimates the full orientation of the device relative to the moving vehicle without relying on any additional hardware or the geomagnetic field by identifying reference directions using knowledge of the vehicle dynamics.
- determines position of the device relative to the vehicle's center of gravity using measurements of the vehicle's pitch and roll dynamics taken at the device. Cross correlation features extracted from the IMU's accelerometer and gyroscope measurements are used as input to a Support Vector Machine to determine if the device is in the driver's seat position and should be deactivated.

Thesis Outline

The remainder of this thesis is organized as follows. In Chapters 2-4, the respiratory sensor application is developed. First in Chapter 2, a novel method of respiratory displacement estimation from IMU measurements is presented, along with experimental results and analysis of factors contributing to displacement estimation error. The respiratory displacements are used in Chapter 3 to further estimate a broad range of respiratory variables related to the timing, phase, and volume of the respiration, and techniques for estimating each type of respiratory variable are developed along with presentation of experimental results. Finally in Chapter 4, experimental results from a recent IRB-approved small-cohort study are presented.

In Chapters 5 and 6, the Food Intake Monitoring (FIM) device is developed. In Chapter 5, methods of detecting if the hand is near the mouth are discussed, beginning with the use of multiple IMUs and following with a simple technique using only a single wrist-worn IMU in conjunction with a permanent magnet. Next, experimental results are presented demonstrating that features extracted from only a wrist worn IMU may be used to classify hand-near-mouth activity as eating or not. In Chapter 6, a method is developed to capture images of the food being consumed using a camera on the wrist-worn device. Finally, a convolutional neural network specially adapted to microcontroller deployment is utilized to further classify the captured images as containing food or not as a means to increase the user-privacy of the device.

In Chapter 7, a vehicular safety application to automatically detect mobile device seat position inside a moving vehicle is developed. The algorithm aims to improve personal safety by providing a means to automatically disable a mobile device when the user is detected to be in the driver's seat position. The first step is to find the orientation of the device relative to the vehicle. Measurements of gravity during periods of low acceleration are used as one reference direction.

Then, measurements of acceleration during breaking are used to find the longitudinal axis of the vehicle. Together, the two reference directions constrain the orientation of the device to the vehicle. Next, vehicle pitch and roll dynamics are presented, and simulations are used to demonstrate how the cross correlation between the vertical acceleration measurements and the gyroscope measurements in the pitch and roll axes can be used to identify which seat the device is in relative to the vehicle center of gravity. Experimental results of real-world driving demonstrate reliable identification of seat position.

Chapter 2:

Estimation of respiratory displacements using IMUs

Introduction

Respiratory variables play a key indicating role in many diseases. For instance, decisions regarding the intubation of a COVID-19 patient on a mechanical ventilator are made in the hospital based on measurement of respiratory rate, tidal volume, and the presence of paradoxical breathing [8]. Long term trends in respiratory variables including forced vital capacity are indicators for neurodegenerative diseases such as Duchenne Muscular Dystrophy, in which respiratory failure develops in advanced stages, requiring eventual mechanical ventilation [9], [10]. Frequent or even continuous monitoring of respiratory variables is needed to help guide clinical interventions in many scenarios, yet there is currently no device available for free-living respiratory monitoring outside of a clinical environment [11].

The most typical method to quantify respiratory parameters in a clinical setting is spirometry, in which the patient breathes into a mouthpiece containing an inline flow or volume measurement sensor [12]. To achieve accurate results, the patient must wear a nose clip and also tightly seal his/her lips around the mouthpiece to ensure that all respiratory flow passes through the spirometer while performing the specified maneuver. The instrumentation itself can lead to patient anxiety and discomfort [13], and thus spirometry does require focus, cooperation, and coaching, which can pose challenges in certain populations, notably children [14].

Alternatively, thoracoabdominal breathing displacement can be used to indirectly estimate respiratory variables. Konno and Mead showed in the 1960s that a two compartment linear model containing anteroposterior surface displacements of the rib cage and abdomen can be used to predict tidal volume [15]. Closely related to the method of measuring anteroposterior displacements, respiratory inductance plethysmography (RIP) measures circumferential changes in two straps worn around the chest and abdomen by measuring inductance changes during the relative motion of conductive threads woven into the straps as they expand and contract during breathing. These changes are then used to predict respiratory variables. One limitation of two degree of freedom approaches such as RIP is that the accuracy is dependent on posture, primarily due to tidal volume dependency on the axial displacements of the chest [11]. The axial displacement of the chest relative to the abdomen can be measured using a magnetometer based displacement sensor, allowing for a three degree of freedom model [16]. In this approach, two

points of the thoracoabdominal wall are outfitted with magnetometer receiver coils, each tuned to particular frequencies to isolate their displacements relative to transmitter coils placed on the patient's back. The most recent version of the displacement based approach, an optical technique called Structured Light Plethysmography (SLP), projects a grid onto the patient's thoracoabdominal wall that is then imaged using video cameras. From these videos, the three dimensional dynamic changes of the entire thoracoabdominal wall can be reconstructed in software and used to estimate volume changes during respiration [17]. The number of surface displacement points that can be analyzed is conceivably limited only by the camera resolution. Displacement measurement using SLP requires an expensive clinical laboratory facility.

In this chapter, we present a novel inertial measurement unit (IMU) respiratory sensor that estimates 3-axes (anteroposterior, inferosuperior, and mediolateral) displacements at multiple points on the thoracoabdominal surface. The sensors are powered by rechargeable batteries and require no real time connection to additional hardware such as a computer or tablet to take measurements. The sensor can be worn on straps around the body or adhered directly to skin using biocompatible tape. The present study uses two sensors, one on the rib cage and one on the abdomen, but the number of sensors is limited only by the size of the sensor itself.

The use of IMUs for attitude and general pose estimation has been of significant interest to the Mechatronics research community. Recent examples of notable papers in this sphere include orientation tracking in the presence of large active accelerations [18], pose estimation using visual-inertial odometry for a smartphone application [19], drift-free 3D orientation estimation combining an IMU with two RTK GPS sensors [20], and IMU-based attitude estimation in the presence of narrow band disturbances [21].

Several inertial-based respiratory monitoring systems have also been previously reported in research literature. The authors in [22] attached a single axis accelerometer to the torso and demonstrated that respiratory rate could be determined by finding the peak of the frequency spectrum of the low-pass filtered acceleration signal. A Kalman filter was employed in [23] to estimate Euler rotation angles of the chest from tri-axis accelerometers and gyroscopes. From the Euler angles, the authors estimated respiration rate using a peak detection algorithm, and demonstrated the algorithm's ability to estimate respiration rate during exercise. Fekr and colleagues found the respiration rate by detecting the number of peaks per minute in the low-pass filtered chest anteroposterior acceleration signal, and they further estimated tidal volume variability by fitting linear trends in the peak-to-valley distance of the normalized anteroposterior acceleration signal for each breath [24]. Siqueira et al. studied the optimal placement of

accelerometers on the thoracoabdominal wall for estimation of the respiratory waveform, finding that accelerometers placed on the right side of the body offer the best performance [25]. This finding is likely due to the larger size of the right lung relative to the left. Instead of estimating specific clinical respiratory variables, the authors found an optimal linear transformation between acceleration measurements and the measurement of a pressure transducer mounted inside a facial mask. They also demonstrated that the pressure transducer reference may not be necessary for finding the optimal linear transformation by modeling the acceleration measurements as a blind source separation problem and extracting the respiratory waveform using independent component analysis.

To the best of our knowledge, our IMU system is unique among previously reported inertial-based systems in that we estimate three dimensional linear surface displacements, akin to the SLP method. The advantage of our approach is that thoracoabdominal displacements in and of themselves may be useful diagnostic measures. For instance, Elsafie and colleagues compared overall chest wall movement, thoracoabdominal displacement asynchrony, and right-left thoracic displacement asynchrony before and after lung resection surgery using SLP [26]. They found that after surgery, the operated side contributed significantly less to the overall chest wall movement as compared to pre-surgery, and there was an increase in the chest wall movement on the non-operated side to compensate. Further, their study found a significant increase in thoracoabdominal displacement asynchrony immediately after surgery. In a systematic review of several studies concerned with the impact of thoracic surgery on chest wall motion, Tukanova et al. suggest that monitoring thoracoabdominal displacements post-surgery could be used to adapt patient-specific physiotherapy [27]. RIP systems can be used to measure thoracoabdominal asynchrony, but the RIP method of measuring circumferential expansion as a proxy for displacement cannot be used for right-left asynchrony analysis. Because it estimates point displacements, an IMU-based system does have the capability to monitor both thoracoabdominal and right-left thoracic asynchrony. Unlike SLP, the IMU-based system has the potential to quantify respiratory kinematics out of clinic after thoracic surgery for longer term patient monitoring in home-based settings.

The remainder of this chapter is organized as follows: Section II presents details of the hardware and software algorithms utilized to estimate the respiratory displacements, Section III presents preliminary experimental results of the sensor demonstrating the feasibility of estimating respiratory displacements with the system, and Section IV concludes the chapter with a discussion of the results and factors contributing to displacement error. The significance of this

chapter is that it is the first reported research work documenting accurate estimation of 3d respiratory displacements using inertial sensors.

Methods

Sensor Hardware Overview

The wearable sensor is shown in Fig. 2. Hardware consists of a 6-axis IMU (TDK Invensense ICM-42605) connected to a microcontroller (Microchip SAMD21) over two wire interface (TWI). The IMU contains a 3-axis accelerometer and a 3-axis gyroscope. Data is stored onboard using microSD flash memory connected via serial protocol interface (SPI) for offline processing. The IMU is configured to sample motion data at a 1 kHz sampling rate using 16-bit analog to digital converters (ADCs), with the measurement range set to ±19.62 m/s^2 for acceleration and ±0.273 rad/s for rotation rate. The hardware is powered using a 750 mAh lithium polymer battery, rechargeable over a universal serial bus (USB) connector. Additional electronics include a power switch and an LED status indicator. The total cost of the system is less than $50.00 USD based on small quantity orders of the requisite components.

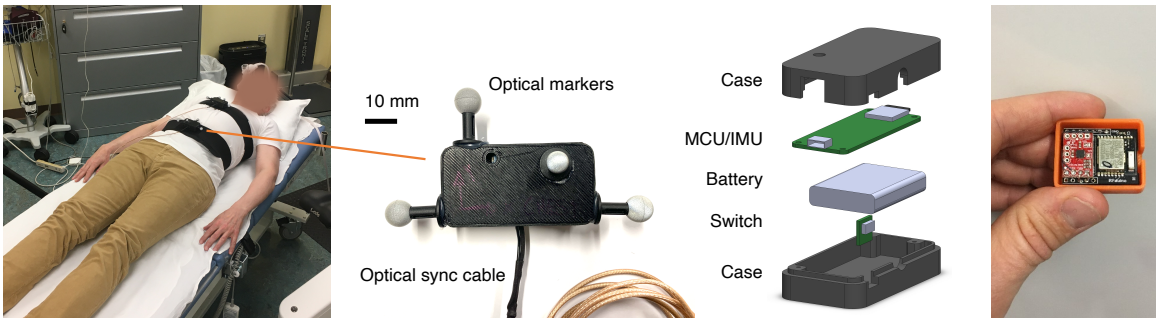

Fig. 2. Sensor hardware and experimental setup
Electronics contain a microcontroller, IMU, and microSD for data logging. The system is powered with a 750 mAh rechargeable Li-Polymer battery. An LED indicates status. The system is assembled in a plastic housing and worn on straps. Displacements estimated with the IMU are compared against an optical motion capture system. A related Bluetooth version of our device shown on right and used in previous work.

The electronics are housed in a 3D-printed enclosure, and we utilize one sensor assembly for each point on the body of interest (e.g. one sensor on the chest, one on the abdomen in this paper). A strip of hook and loop tape on the back of the enclosure allows the sensor assembly to be easily attached to an elastic body-worn strap. A coaxial cable is used to synchronize the wearable sensors with an optical motion capture system used for ground truth displacement measurement.

Algorithm overview

A typical approach towards displacement estimation would consist of double integrating IMU acceleration measurements during breathing to estimate thoracoabdominal surface displacements at each sensor location, as shown in the lower part of the high level block diagram in Fig. 3. However, this technique is practically challenging for two reasons: (a) respiration displacements are relatively small, less than ±10 mm, and occur at low fundamental frequencies in the range of 0.2 to 0.5 Hz for normal breathing in adults and (b) MEMS accelerometers measure not only the linear accelerations from breathing but also gravity, which constitutes a much larger component. For the purposes of this work, we model the three axis accelerometer measurements as follows:

$$\boldsymbol{y}_{acc}(t) = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \boldsymbol{a}_{lin}(t) - \boldsymbol{g}_S(t) + \boldsymbol{b}(t) + \boldsymbol{v}(t) \tag{1}$$

Here, $\boldsymbol{a}_{lin}(t)$ is the external or linear acceleration of the sensor caused by respiration, $\boldsymbol{g}_S(t)$ is the gravity vector in the sensor frame of reference, $\boldsymbol{b}(t)$ is a possibly time varying offset bias term, and $\boldsymbol{v}(t)$ is additive white gaussian noise. Essentially, the objective of the algorithm is to isolate $\boldsymbol{a}_{lin}(t)$ from the other components in the acceleration measurement and double integrate it to obtain displacement. The components of $\boldsymbol{a}_{lin}(t)$ can have magnitudes of interest down to hundreds of μg for the slowest rates and displacements of interest, and so the magnitude of $\boldsymbol{y}_{acc}(t)$ is dominated by the 1 g gravity term.

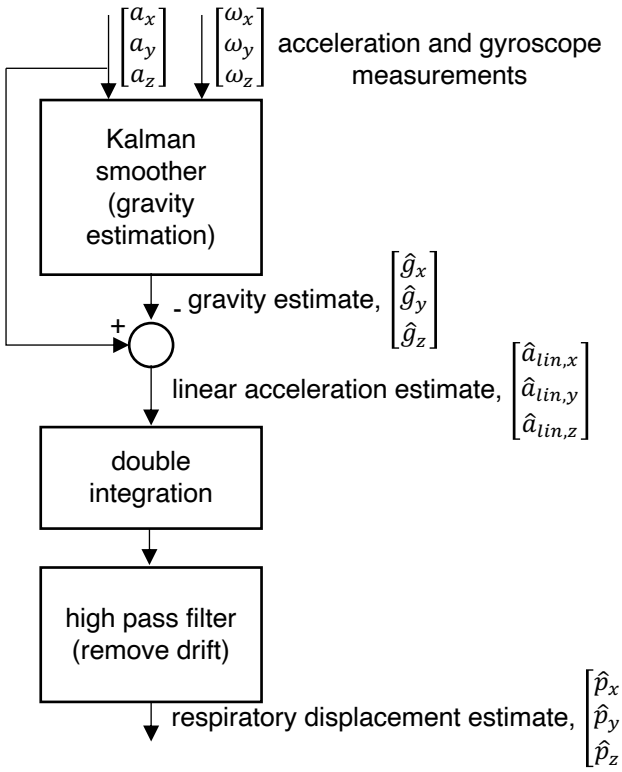

Fig. 3. Algorithm block diagram
A Kalman smoother estimates the gravity vector. The acceleration residual is double integrated and high pass filtered.

The presence of the bias term $\boldsymbol{b}(t)$ leads to drift when double integrating the raw acceleration measurement, which is shown in Fig. 4 on a set of experimental data. However, in practice, $\boldsymbol{b}(t)$ varies at a frequency much lower than the frequencies typical of respiration, and so this term can be removed using a high pass filter with a cutoff frequency lower than all expected frequencies of breathing. The removal of drift using high pass filtering is shown in Fig. 5. There is definite improvement in the estimate using high pass filtering, but still there is significant disagreement between the IMU estimate and the ground truth reference. This remaining disagreement can be explained by the presence of gravity in the acceleration measurement.

If the sensor experienced no or only very slow rotations during breathing, then the $\boldsymbol{g}_S(t)$ term would be slowly time varying at frequencies much lower than the fundamental frequency of the breathing motion and could be removed from the acceleration measurement with a high pass filter in the same manner as removing the bias term. But if $\boldsymbol{g}_S(t)$ is periodic and in the same frequency range as $\boldsymbol{a}_{lin}(t)$, then the $\boldsymbol{g}_S(t)$ term cannot be isolated from $\boldsymbol{a}_{lin}(t)$ using frequency-based filtering. In fact, we find the latter situation to be the case, as our optical motion capture

system shows the torso undergoes subtle bending primarily in the sagittal plane on the order of 0.03 rad (2°) during each breath. A visualization of this rotation is shown in Fig. 6. Because of the frequency overlap of $\boldsymbol{a}_{lin}(t)$ and $\boldsymbol{g}_S(t)$, we utilize a model-based approach to estimate $\boldsymbol{g}_S(t)$, using a gyroscope to measure the angular velocity of the sensor. This approach requires no prior assumption about the rates of rotation relative to breathing. Because offline analysis can be utilized in our application, we utilize a Kalman smoother to estimate $\boldsymbol{g}_S(t)$, which is presented in the next sections.

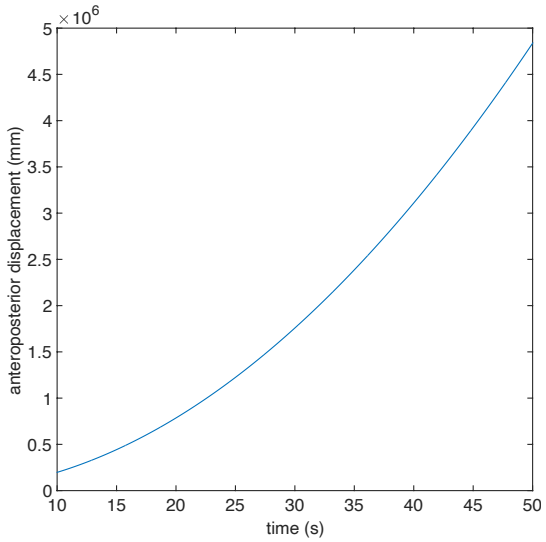

Fig. 4. Drift from simple double integration of the acceleration

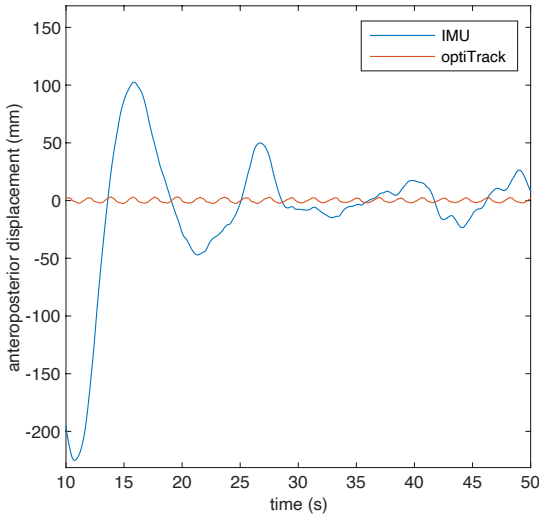

Fig. 5. Drift removal via high pass filter
Drift is removed after high pass filtering of the double integrated acceleration measurement, but there is still disagreement between the IMU estimate and the reference optical system.

Kinematics of the gravity vector

First, define R_{SE} as the orthonormal rotation matrix ($R_{SE} \in SO(3) = \{R \in \mathbb{R}^{3\times3} | RR^T = I, detR = 1\}$) which transforms a vector $\boldsymbol{x}_E \in \mathbb{R}^3$ expressed in the earth frame to its representation in the sensor frame, $\boldsymbol{x}_S \in \mathbb{R}^3$:

$$\boldsymbol{x}_S = R_{SE}\boldsymbol{x}_E \tag{2}$$

The rotation matrix R_{SE} is equivalently referred to as the Direction Cosine Matrix (DCM) and the Attitude Matrix. In a dynamic situation in which the sensor's orientation changes relative to earth frame E, the DCM is time varying. Of particular interest in this application is the gravity vector $\boldsymbol{g}$, which is fixed in direction relative to earth and has a magnitude of 1 g. That is, the vector $\boldsymbol{g}_E$ is not time varying, but as the sensor undergoes rotation, $\boldsymbol{g}_S$ (gravity expressed in the sensor frame) will vary with time and so:

$$\boldsymbol{g}_S(t) = R_{SE}(t)\boldsymbol{g}_E \tag{3}$$

Next, take the time derivative of (3), which gives the following:

$$\dot{\boldsymbol{g}}_S(t) = \dot{R}_{SE}(t)\boldsymbol{g}_E \tag{4}$$

The DCM R_{SE} has its own kinematic equation (see [5] for derivation):

$$\dot{R}_{SE}(t) = -[\boldsymbol{\omega}_S(t)]_\times R_{SE}(t) \tag{5}$$

Here, $\boldsymbol{\omega}_S(t) \in \mathbb{R}^3$ is the angular velocity of the sensor relative to earth, expressed in the sensor frame. The notation $[\boldsymbol{a}]_\times$ indicates the skew symmetric cross product matrix ($[\boldsymbol{a}]_\times \in \{A \in \mathbb{R}^{3\times3} | A^T = -A\}$), defined as follows for $\boldsymbol{a} \in \mathbb{R}^3 = [a_1 \quad a_2 \quad a_3]^T$:

$$[\boldsymbol{a}]_\times \equiv \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \tag{6}$$

Utilize equation (5) to replace the $\dot{R}_{SE}(t)$ term in equation (4):

$$\dot{\boldsymbol{g}}_S(t) = -[\boldsymbol{\omega}_S(t)]_\times R_{SE}(t)\boldsymbol{g}_E \tag{7}$$

Next, notice that $R_{SE}(t)\boldsymbol{g}_E$ is equivalent to $\boldsymbol{g}_s(t)$, which simplifies the kinematic equation to its final form:

$$\dot{\boldsymbol{g}}_S(t) = -[\boldsymbol{\omega}_S(t)]_\times \boldsymbol{g}_S(t) \tag{8}$$

Equation (8) is the kinematic differential equation for the gravity vector in the sensor frame and shows how the time evolution of the gravity vector depends on the angular velocity of the sensor. With the assumption of constant angular velocity between samples k and $k+1$ with sample time T, we can discretize the kinematic equation for the gravity vector in the body frame (8) as follows:

$$\boldsymbol{g}_S(k+1) = e^{-T[\boldsymbol{\omega}_S(k)]_\times}\boldsymbol{g}_S(k) \tag{9}$$

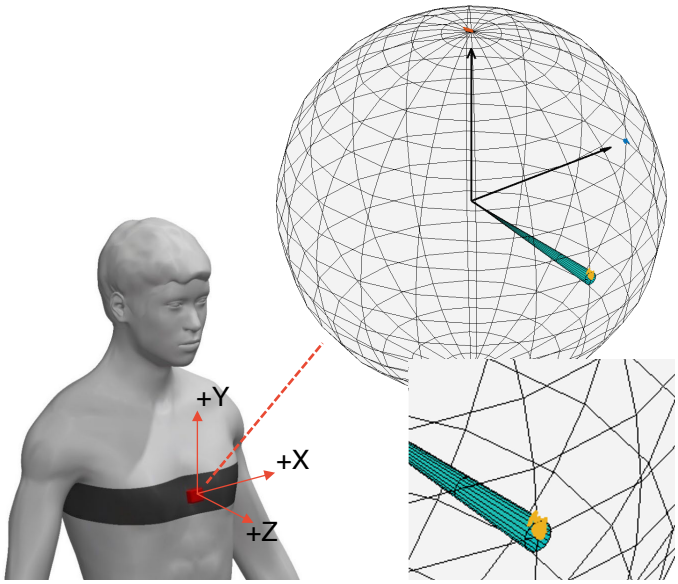

Fig. 6. Respiratory rotations
Green cone (0.03 rad, 2° angle) plotted showing rotations that primarily occur about the X direction during a respiration experiment, as recorded by an optical motion capture system. Rotations are small, but they occur at frequencies at or near the fundamental respiratory frequency.

Measurement model for gravity estimation

As discussed above, the accelerometer measurement magnitude is dominated by gravity. For the purpose of estimating the gravity vector $\boldsymbol{g}_S$, we can thus reasonably ignore the linear acceleration term which is more than 1 order of magnitude smaller. We will also assume the bias term $\boldsymbol{b}(t)$ is constant and has been removed, since it is two orders of magnitude smaller after initial calibration. This leaves a simple accelerometer measurement model for the gravity estimation step:

$$\begin{aligned} \boldsymbol{y}_{acc}(k) &= -\boldsymbol{g}_S(k) + \boldsymbol{v}(k), \\ \boldsymbol{v}(k) &\sim \mathcal{N}(0, \sigma_{acc}^2 I_3) \end{aligned} \tag{10}$$

Error state Kalman framework for gravity estimation

Together, Equations (9) and (10) constitute the basis for estimating the gravity vector using a Kalman smoother. One issue with using the gravity vector directly as the state vector concerns the Kalman correction step. Note that $\boldsymbol{g}_s$ has three elements but only two degrees of freedom (tilt and roll angles). Given any two elements of $\boldsymbol{g}_s$, the third element can be determined by enforcing the 1 g constraint. Ideally, the gravity estimator should respect this constraint. However, consider the measurement update during the Kalman correction step at time step k:

$$\hat{\boldsymbol{x}}_k^+ = \hat{\boldsymbol{x}}_k^- + K(\boldsymbol{y}_k - \hat{\boldsymbol{y}}_k) \tag{11}$$

Here, $\hat{\boldsymbol{x}}_k^-$ and $\hat{\boldsymbol{x}}_k^+$ are the prior and posterior estimates, respectively, K is the Kalman gain, $\mathbf{y}_k$ is the measurement and $\hat{\mathbf{y}}_k$ is the predicted measurement. There is no unique property of the $K(y_k - \hat{y}_k)$ additive update term which guarantees the posterior estimate has a magnitude of 1 g, even if the prior estimate does have a constrained magnitude, and so our estimated gravity vector is unlikely to retain this constraint over time.

This magnitude constraint issue is sometimes handled by renormalizing the state vector after each time update (see, for instance [28]), but we find better results in practice by adopting an indirect or error state framework. In this framework, the smoother separately tracks a global state and also a small rotation error state of local uncertainty around the global state. For a complete introduction to error state Kalman filters, see Chapter 6 of [5]. Let $\boldsymbol{g}_S$ be the true gravity vector in the sensor frame (the aforementioned global state), and let $\hat{\boldsymbol{g}}_S$ be the gravity vector estimate in the sensor frame. Assuming both $\boldsymbol{g}_S$ and $\hat{\boldsymbol{g}}_S$ to be unit vectors, then they are related by some small error rotation from the estimated to true sensor frames, defined as $R_{S\hat{S}}$:

$$\boldsymbol{g}_S = R_{S\hat{S}}\hat{\boldsymbol{g}}_S \tag{12}$$

As any rotation matrix has three degrees of freedom, $R_{S\hat{S}}$ can be parameterized with a three component vector, $\boldsymbol{\delta v} \in \mathbb{R}^3$, the aforementioned error state. Define the parameterization of the rotation matrix $R_{S\hat{S}}$ as follows:

$$R_{S\hat{S}}(\boldsymbol{\delta v}) \equiv e^{-[\boldsymbol{\delta v}]_\times} \approx I_3 - [\boldsymbol{\delta v}]_\times \tag{13}$$

In fact, the error state is overparameterized in this case, as any correction to $\hat{\boldsymbol{g}}_S$ can be reached by rotating about an axis constrained to the plane normal to $\hat{\boldsymbol{g}}_S$. However, the motivation for this application is maintaining the estimate norm constraint as discussed above, rather than providing the computational benefits of a minimal error-state parameterization.

The kinematic equation of the gravity vector estimate $\hat{\boldsymbol{g}}_S$ is identical to the true kinematic equation, with the exception that the true body angular velocity $\boldsymbol{\omega}_S$ is replaced with the gyroscope measurement, $\boldsymbol{y}_\omega$:

$$\dot{\hat{\boldsymbol{g}}}_S(t) = -[\boldsymbol{y}_\omega(t)]_\times \hat{\boldsymbol{g}}_S(t) \tag{14}$$

We assume the gyroscope measurement is corrupted by zero mean gaussian white noise:

$$\boldsymbol{y}_{\omega}(t) = \boldsymbol{\omega}_S(t) + \boldsymbol{w}(t) \tag{15}$$

Taking the time derivative of (12) and substituting into (8) along with the gyroscope measurement model gives the following:

$$\dot{R}_{S\hat{S}} = -[\boldsymbol{y}_{\omega}]_{\times} R_{S\hat{S}} + [\boldsymbol{w}]_{\times} R_{S\hat{S}} + R_{S\hat{S}}[\boldsymbol{y}_{\omega}]_{\times} \tag{16}$$

Substituting the first order error approximation in (13) and ignoring small products of the noise term with the error vector gives the following equation:

$$\boldsymbol{\delta\dot{v}} = -[\boldsymbol{y}_{\omega}]_{\times}\boldsymbol{\delta v} - \boldsymbol{w} \tag{17}$$

Equation (17) is used to propagate the state covariance, P, in the smoother prediction step.

Next, consider the accelerometer measurement model in (10) in terms of the error state and its first order approximation from equation (13):

$$\begin{gathered}\boldsymbol{y}_{acc} = -R_{S\hat{S}}(\boldsymbol{\delta v})\hat{\boldsymbol{g}}_S + \boldsymbol{v} \approx -\hat{\boldsymbol{g}}_S - [\hat{\boldsymbol{g}}_S]_{\times}\boldsymbol{\delta v} + \boldsymbol{v}, \\ \boldsymbol{v} \sim \mathcal{N}(0, \sigma_{acc}^2 I_3)\end{gathered} \tag{18}$$

Thus, in terms of the error state, the sensitivity matrix to use in the Kalman gain and covariance calculations in the measurement correction step is:

$$H\left(\widehat{\boldsymbol{\delta v}}^{-}\right) \approx -[\hat{\boldsymbol{g}}_S]_{\times} \tag{19}$$

The measurement update will produce a non-zero estimate of the error state using the typical additive Kalman update:

$$\widehat{\boldsymbol{\delta v}}^{+} = \widehat{\boldsymbol{\delta v}}^{-} + K\left(\boldsymbol{y}_{acc} + e^{-[\widehat{\delta v}^{-}]_{\times}}\hat{\boldsymbol{g}}_S^{-}\right) \tag{20}$$

The measurement correction to the error state suggests that the best estimate of $\boldsymbol{g}_s$ should now take the updated error state into account. This so-called reset operation pushes the error state estimate to the global estimate and resets the error to zero prior to the next dynamic prediction step:

$$\hat{\boldsymbol{g}}_S^{+} = R(0)\hat{\boldsymbol{g}}_S^{+} = e^{-[\widehat{\delta v}^{+}]_{\times}}\hat{\boldsymbol{g}}_S^{-} \tag{21}$$

$$\widehat{\boldsymbol{\delta v}}^{++} = 0 \tag{22}$$

The reset is sometimes accompanied by a corresponding modification to the covariance. However, we have not implemented this step as some authors have observed no significant impact in practical application when this step is ignored [29].

Simulations show that the error-state filter presented above results in better estimates of gravity compared to a renormalization approach. In the renormalization approach, the gravity estimate is directly updated using the standard Kalman additive update (11). As this update is not constrained, the renormalization approach enforces a gravity magnitude constraint of 1 g by normalizing the posterior estimate each time step. A series of simulations were run in which the ground truth gravity vector was randomly initialized. To simulate acceleration measurements typically encountered during breathing, a small sinusoidal rotation of 0.035 rad (2º) amplitude at 0.5 Hz was overlaid with a 1 mg sinusoidal external acceleration at the same frequency as rotation. The simulated accelerometer and gyroscope measurements were additionally corrupted with zero mean gaussian noise with variances randomly chosen and ranging from 1e-5 and 1e-2. The error-state filter outperformed the renormalization filter in 65% of simulations, with max improvement in RMS error of just over 0.5%.

Error state fixed interval Kalman smoother for gravity estimation

The estimation of the gravity vector could be accomplished using a Kalman filter and the state and measurement equations presented above. However, we find that the filter estimate will exhibit a phase lag in some instances. This lag can be seen by finding the cross correlation between the gravity estimate and the acceleration measurement, shown in Fig. 7 for one particular experiment. When the gravity estimate is removed from the acceleration measurement, this phase lag leads to incomplete cancellation of the gravity vector from the acceleration measurement. The remaining residual from the incomplete cancellation will be a signal at near the same frequency as the respiratory signal. Because the linear accelerations are often much smaller than the signal variation due to gravity changes, the residual from incomplete gravity cancellation can be on the same order of magnitude as the linear accelerations. The lag can be removed and the gravity estimate improved by using a smoother as opposed to a filter, as shown in Fig. 7. A smoother cannot be employed real-time in closed-loop control systems, but offline analysis is satisfactory for medical diagnostics. Essentially, the smoother is “delayed real-time,” with the delay governed by the chosen smoother interval (60-120 s). After the data collection interval has passed, displacement estimates are immediately available to the physician.

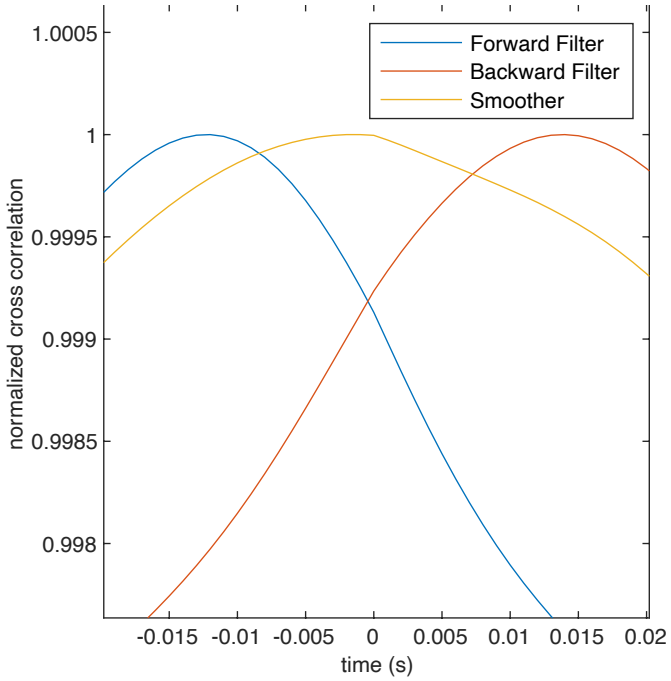

Fig. 7. Filter and smoother lags
Experimental comparison of the cross correlation between the gravity estimate and the acceleration measurement in the anteroposterior direction. The forward and backward filters have phase lags in opposite time directions, which causes incomplete gravity cancellation.

There are several Kalman smoother variants, but here we employ a fixed interval smoother, in which all measurements in the entire measurement interval are used to determine the estimate at all points in the interval. Essentially, we run two versions of the Kalman Filter. The first filter runs in the forward time direction. The second filter runs in the backward time direction for the measurement interval. Then, at each point in time, the forward and backward filters are optimally combined. The optimal smoother covariance can be shown to be [30]:

$$P^s = \left(P^{f\,-1} + P^{b\,-1}\right)^{-1}, \tag{23}$$

where P^s, P^f, P^b are the smoother, forward filter, and backward filter covariances, respectively. In the scalar case, if the forward and backward filters have identical state estimate variances, the smoother will have half of the variance of the filters (e.g. the smoother outperforms the filters). In the traditional smoother, the filter covariances are used to weight the forward and backward estimates to produce the smoother estimate in the following manner:

$$\hat{\boldsymbol{x}}_k^S = P^s \left({P_{k|k}^f}^{-1} \hat{\boldsymbol{x}}_{k|k}^f + {P_{k|k+1}^b}^{-1} \hat{\boldsymbol{x}}_{k|k+1}^b \right), \tag{24}$$

where the s, f, and b superscripts indicate smoother, forward, and backward covariances and estimates. The time indices indicate that we incorporate the measurement at time k in the forward filter but not the backward filter, since we can only use a single measurement one time.

As before, the problem with this approach is the 1 g constraint in the gravity vector. This can be easily seen with an example. Suppose the forward estimate of gravity is $\hat{\boldsymbol{g}}_{k|k}^f = [1 \quad 0 \quad 0]^T$, while the backward estimate is $\hat{\boldsymbol{g}}_{k|k+1}^b = [0 \quad 1 \quad 0]^T$. Suppose also that the forward and backward covariances are identical, $P^f = P^b$. Then the smoother estimate using (24) would be $\hat{\boldsymbol{x}}_{k|k}^f = [0.5 \quad 0.5 \quad 0]^T$. However, this is not a valid estimate of the gravity vector, since its norm is not 1 g.

To solve this issue, we follow a similar approach as outlined in [31] for preserving unit-norm quaternion estimates in a Kalman smoother, applied to gravity vector estimation. Let $\boldsymbol{g}^f$ and $\boldsymbol{g}^b$ be the forward and backward gravity estimates, respectively, and define an error vector $\Delta \boldsymbol{g}$ as follows:

$$\Delta \boldsymbol{g} = \boldsymbol{g}^f \times \boldsymbol{g}^b \tag{25}$$

The $\Delta \boldsymbol{g}$ term represents a small rotation from $\boldsymbol{g}^f$ to $\boldsymbol{g}^b$, since the cross product vector points in the normal direction of the plane containing $\boldsymbol{g}^f$ and $\boldsymbol{g}^b$, with a magnitude equal to the sine of the angle between them. We finally apply a portion of the $\Delta \boldsymbol{g}$ rotation to $\boldsymbol{g}^f$ depending on the relative covariances of the forward and backward estimates. In a similar manner as in [31], we weight Δg as follows:

$$\boldsymbol{\delta g} = P^s {P^b}^{-1} \Delta \boldsymbol{g}, \tag{26}$$

where P^s is calculated using (23). The smoother correction rotation is applied to the forward estimate using an axis angle pair, with the axis of rotation defined as the unit vector in the direction of $\boldsymbol{\delta g}$, and the sine of the angle as its magnitude. The correction is straightforward to implement using Rodrigues' rotation formula [32]:

$$\boldsymbol{g}^s = \left(I_3 + [\boldsymbol{\delta g}]_\times + \sqrt{1 - \|\boldsymbol{\delta g}\|_2^2} \left[\frac{\boldsymbol{\delta g}}{\|\boldsymbol{\delta g}\|} \right]_\times^2 \right) \boldsymbol{g}^f \tag{27}$$

When the backward estimate has large uncertainty relative to the forward estimate, the smoother correction gives the forward estimate, and vice versa. With similar levels of uncertainty, the smoother applies a correction in between the forward and backward estimates. A summary of the Kalman smoother equations is shown in Fig. 8. A particular example showing the improvement in displacement estimation using the smoother is shown in Fig. 9.

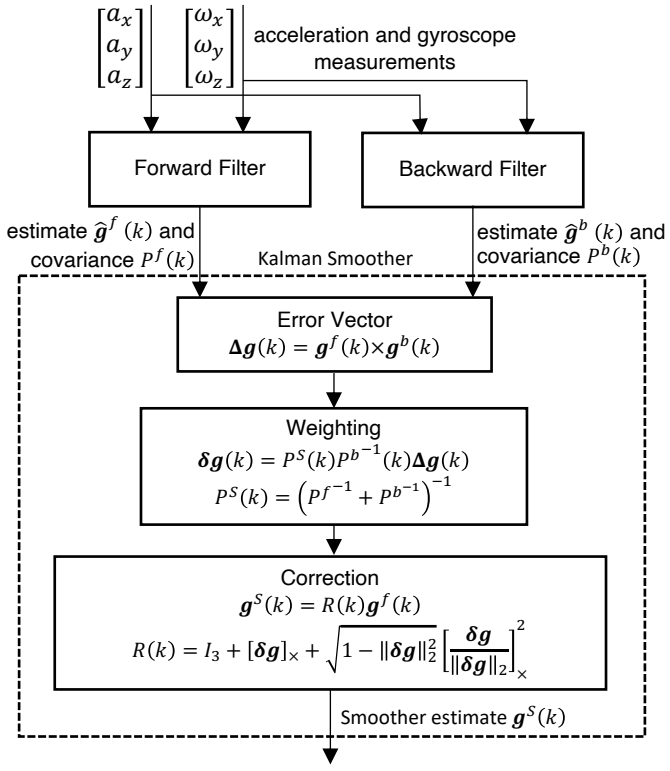

Fig. 8. Smoother block diagram and equations.

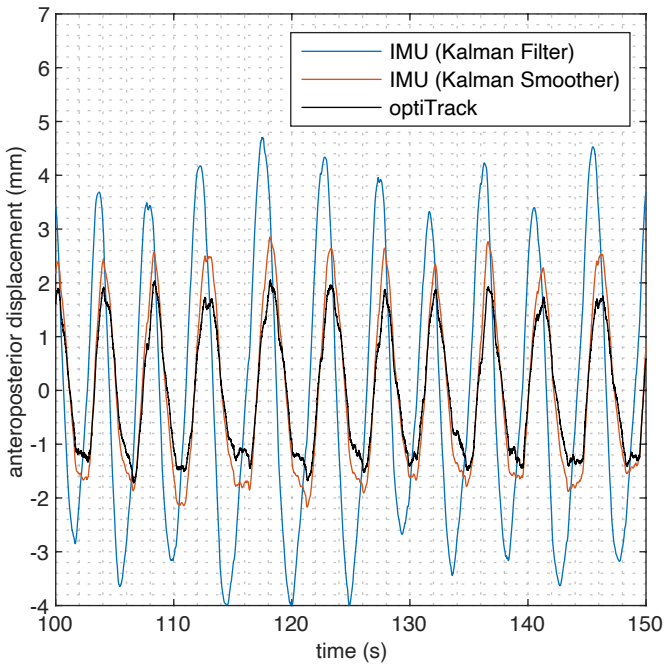

Fig. 9. Kalman smoother improvements
Experimental example showing improvement in the displacement estimate using the Kalman smoother.

Double integration and high pass filtering

After estimating the gravity vector using the Kalman smoother, the residual of the acceleration measurement and the gravity estimate is double integrated to obtain position. Since there is always some remaining bias in the acceleration measurement, the double integrated acceleration will drift, which can be removed with a high pass filter. Since we are utilizing offline analysis in this application, we use forward-backward filtering (Matlab "filtfilt"), which produces a drift-free estimate of respiratory displacements with no phase lag. Through an iterative process, a 6th-order Butterworth filter with frequency cutoff of 0.07 Hz was found to provide reliable drift removal. The entire algorithm is implemented using Matlab and the Signal Processing Toolbox.

Results

Preliminary testing of the respiratory monitoring device was conducted using two sensors, one on the chest worn at the approximate midpoint of the sternum, and one worn at the umbilicus. The results presented here were obtained on a single subject during five experiments in a variety of body postures and respiratory rates (see Table 1). The IMUs were visually aligned with body axes such that the IMU x axis was aligned in the body left lateral direction, IMU y in the body superior direction, and IMU z in the anterior direction.

TABLE 1: EXPERIMENTAL PROTOCOL				
ID	Posture	Duration (s)	Breaths	Experiment Details
00	standing	180	37	Normal breathing
01	standing	96	14	Deliberate large breaths
02	sitting	272	56	Normal breathing
03	sitting	240	95	Hypo- and hyper-ventilation
04	supine	316	54	Normal breathing

The displacement estimates were compared against an infrared optical motion tracking bar (optiTrack V120:Trio) which has sub-mm accuracy and samples at 120 frames per second. The two systems were synchronized using the V120:Trio sync output, which sends a high signal during recording and a low signal otherwise. The sensors were configured to start and stop data collection using a hardware interrupt triggered by the sync signal. The 1 kHz sampling rate of the IMU is higher than required to capture the frequencies of interest, but there is no downside to a higher sampling rate other than larger memory usage.

A typical example of the device performance at a normal breathing rate near 0.5 Hz is shown in Fig. 10. Examples of breathing during hyperventilation and hypoventilation are shown in Fig. 11. The optical camera system and the IMU system operate at different sampling rates, and the external synchronization trigger of the V120:trio is not designed to be a precision signal, meaning there may be some delay of roughly a few hundredths of a second between the two systems. As a result, we used the peak displacement of each breath to quantify the accuracy of the displacements based on inertial measurements as compared to the optiTrack. IMU versus optiTrack displacements are plotted in Fig. 12. For all experiments shown in Fig. 12, the overall RMS error was 0.4 mm, with a max error of 1.3 mm. Each plotted breath is labelled according to the frequency of breathing and if the displacement occurred in a vertical or horizontal direction. These factors play a role in the system error, which we discuss in detail in the next section.

The displacement estimates may also be used to determine the respiratory rate by finding the time between detected peaks in a single axis of the displacement estimates. An example of respiratory rate estimated using the displacements from the IMU is shown in Fig. 13 and compared to the rate reported by a clinical spirometer.

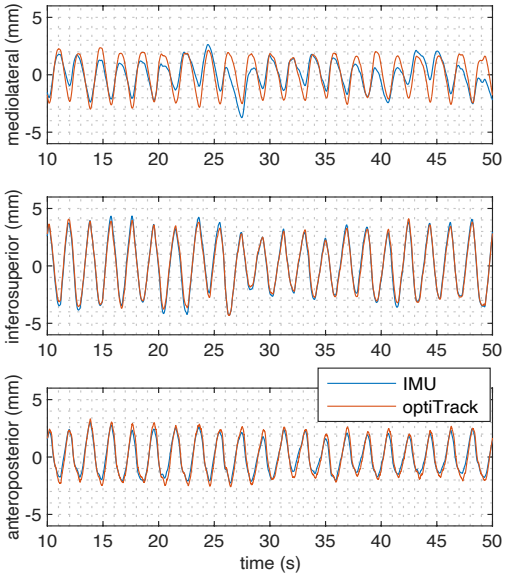

Fig. 10. Representative displacement estimate of the chest
Data was taken during a portion of an experiment of normal breathing while sitting. The inferosuperior direction is nominally aligned with gravity and has the least error.

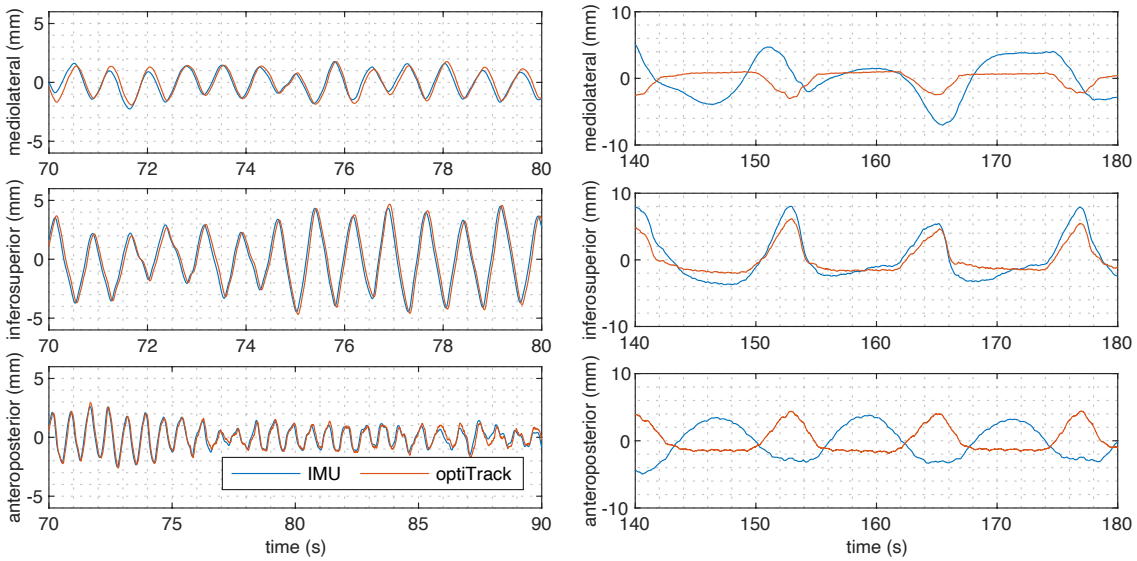

Fig. 11. Influence of respiratory rate and axis orientation
Representative comparison of the influences of respiratory rate and sensor axis orientation on displacement estimation accuracy. At above normal breathing frequency (left plot, $f_0 \approx 1.4\,Hz$), the displacement accuracy is excellent in all axes. At very low respiratory rates (right plot, $f_0 \approx 0.08\,Hz$), the estimation accuracy degrades, particularly in the anteroposterior and mediolateral directions, which are nominally perpendicular to gravity in these experiments.

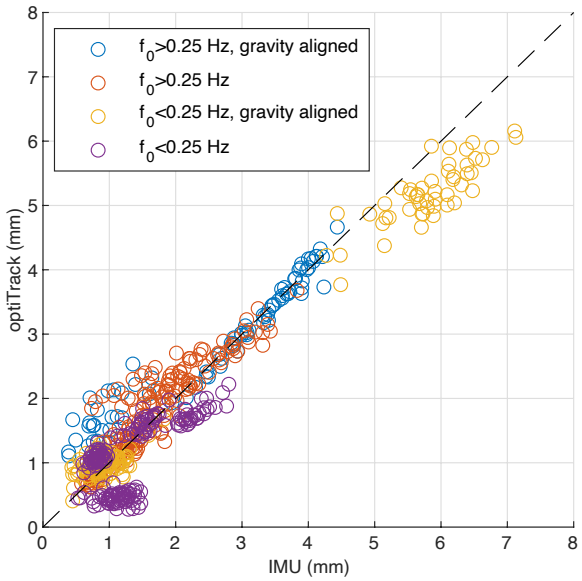

Fig. 12. Optical vs. IMU displacement estimates
Recorded optiTrack peak displacement versus IMU-based estimate for a single participant. Breaths are labeled based on respiratory frequency and the axis alignment with gravity.

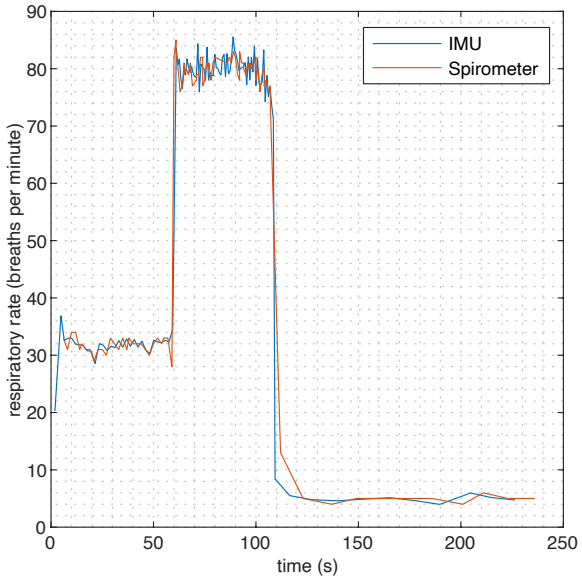

Fig. 13. Respiratory rate estimation
Example respiratory rate estimation using IMU displacement estimates compared to a spirometer.

Discussion

With a max error in the set of experiments as 1.3 mm when treating the optiTrack system as ground truth, the error in the IMU system is of the same order of magnitude as the error in the optical system. Even so, there are subtle differences in accuracy based on two parameters: (1) respiration rate, and (2) the nominal orientation of the sensor relative to gravity. The error was lowest when the displacement was above 2 mm, at a fundamental frequency above 0.25 Hz, and aligned with gravity (RMS error 0.13 mm, max error 0.50 mm). When the displacement was greater than 2 mm and aligned with gravity but at a lower rate than 0.25 Hz, the RMS error was 0.71 mm with a max error of 1.2 mm. When the displacement was greater than 2 mm at a frequency above 0.25 Hz but not aligned with gravity, the RMS error was 0.24 mm with a max error of 0.69 mm. Below 2 mm displacements, these differences are less apparent. Because the achieved accuracy of the IMU system is close to that of the optical reference system, it is difficult to make strong conclusions concerning the errors. However, the differences due to respiratory rate and gravity alignment have straightforward physical explanations.

For the purposes of discussion, consider the breathing signal as a simple sinusoid with magnitude A and fundamental frequency f_0. Normal breathing ranges from a fundamental frequency of approximately 0.25Hz at the lower end in adults and up to 1 Hz in children, with higher frequencies associated with acute respiratory dysfunction [33], [34]. Modeled as a simple sinusoid, the breathing displacement would then follow

$$x = A\,cos(2\pi f_0 t) \tag{28}$$

After taking the second derivative, the acceleration due to this breathing displacement is:

$$a = -4\pi^2 f_0^2 A\,cos(2\pi f_0 t) \tag{29}$$

Because of the f_0^2 term in the acceleration amplitude, as the frequency decreases, the acceleration of respiration at 0.2 Hz will be two orders of magnitude smaller than respiration at 1 Hz, with displacement magnitudes equated. A breathing displacement of 5mm at 0.2 Hz would have an acceleration amplitude of 8.1e-4 g, compared to 2.0e-2 g for 1 Hz respiration. Thus, the narrow band of respiration frequencies and displacements of interest will produce orders of magnitude different accelerations.

Next, consider how the sensor axis orientation relative to gravity affects system accuracy. When a sensor axis is nominally vertical, such as the anteroposterior direction when supine or the inferosuperior direction when standing, that axis is relatively insensitive to changes in gravity as compared to the other two axes. This can be seen by looking at the sensitivity matrix in (19). As

an axis contains most or all of the gravity estimate, that axis has little to no sensitivity to the error in the gravity estimate. This can also be seen in the two dimensional case, by considering a small periodic rotation, $\theta(t)$, around a horizontal axis alongside the linear respiratory displacement. The acceleration measurements perpendicular to the rotation axis in the vertical and horizontal directions, respectively, will be:

$$a_{vert}(t) = a_{lin}(t) + cos\big(\theta(t)\big) \approx a_{lin}(t) + 1 - \frac{\theta^2(t)}{2} \tag{30}$$

$$a_{horiz}(t) = a_{lin}(t) + sin\big(\theta(t)\big) \approx a_{lin}(t) + \theta(t) \tag{31}$$

When θ is near zero, the sensitivity of the vertical acceleration from variation in θ is much less than the sensitivity of the horizontal acceleration. As a result, the residual error from cancelling the gravity signal using the gravity estimate plays a more critical role in horizontal directions than vertical directions. Because of the impact of axis alignment relative to gravity on the displacement estimates, we choose to use the axis nominally aligned with gravity when estimating the respiratory rate as in Fig. 13, as the displacement estimate is generally best in that direction across all respiratory rates.

Given that previous research has shown that the anteroposterior displacements are the most important factors relating thoracoabdominal displacements to tidal volume, a reasonable conclusion is that our system will have the best performance when the user is in the supine position. Automated discrimination between supine, sitting/standing, and other positions while lying can be accomplished by examining which axis contains the bulk of the 1 g gravity component.

The accuracy of the system demonstrated in this paper is comparable to that of the RIP and SLP systems discussed in the introduction, while providing advantages compared to both. The RIP systems provide only circumferential displacements, while the SLP systems can only be utilized in an expensive clinical laboratory. The 3-dimensional wearable displacements measured by the IMU system in this paper can be utilized for long-term monitoring at home, and can measure respiratory rate, tidal volume and thoracoabdominal asynchrony.

Conclusion

Respiratory variables including respiratory rate, tidal volume, and forced vital capacity play an important indicating role in many acute and neurodegenerative diseases. Typical methods of respiratory variable estimation include spirometry, structured light plethysmography, and

respiratory inductance plethysmography. In general, these methods are limited in their applicability to long-term and out-of-clinic monitoring due to a combination of expensive, specialized, or large equipment, and required training for operation. Some methods such as spirometry are not suited for children, as they require comprehension and cooperation, breathing into a mouthpiece, and wearing a nose clip.

In this chapter of the thesis, we have presented a novel IMU-based respiration monitor that overcomes many of these challenges. It is portable, can measure point displacements, and does not require nose clips or masks. Previous efforts at IMU-based respiration monitoring have primarily focused breathing rate estimation. We have demonstrated a novel method of extracting respiratory displacements with accuracy comparable to an optical motion capture system.

The experiments here were performed on a single healthy subject. Thus, the study is limited in understanding how well the algorithm generalizes to a broader population with varying age, body morphology, and gender. Although displacement can be related to respiratory variables such as breath volume, we are unable to say from the work in this chapter how the displacement error affects estimation of other variables. In the subsequent chapters of this thesis, the relationship of displacements found with our system to a variety of respiratory variables will be studied in a small cohort IRB-approved study.

Chapter 3:
Estimation of temporal, phasic, and volumetric respiratory variables using IMUs

Introduction

The previous chapter focused on estimating respiratory displacements using IMUs. In the present chapter, we turn our attention to further estimation of a broad suite of respiratory variables. Measurement of respiratory variables plays a fundamental role in the diagnosis and monitoring of many diseases. For example, the volume of air that a patient can exhale in one second (FEV_1) declines as one ages through adulthood. Patients suffering from chronic obstructive pulmonary disease (COPD) will face increasingly rapid decline in FEV_1 that depends on disease severity, with FEV_1 as low as 30% that of a person with normal lung function and similar age [35]. In the COVID-19 pandemic, mechanical ventilatory support decision criteria include persistently high respiratory rate and abnormal thoracoabdominal wall motions [36]. In neuromuscular diseases such as Duchenne muscular dystrophy, long term muscular decline leads to respiratory failure and mechanical ventilation [9]. The relative contribution and synchrony of the chest, abdomen, and left/right compartments within the chest can be used as markers to tailor physiotherapy in certain post-surgical situations such as lung resection [26]. Thus, there is a clear need for tools that provide information regarding the volume, timing, and physical motions of respiration across a range of acute and chronic diseases.

As discussed in the previous chapter, there are several tools available to measure and estimate such variables, including spirometry, Respiratory Inductance Plethysmography (RIP), and Structured Light Plethysmography (SLP). All of the existing respiratory analysis tools lack some combination of portability and the range of variables they can measure. For instance, SLP requires a specialized laboratory, and spirometry (even while portable) can only measure variables related to the forced expiratory flow and volume. RIP is a portable tool, but it is generally limited to two circumferential measurements. There is currently no existing sensor on the market suitable for long-term home-based respiratory monitoring that can estimate the full

suite of respiratory variables related to timing, multi-compartment phase, volume, and three-dimensional point displacement [11].

In this chapter of the thesis, we present preliminary experimental results from the novel IMU-based respiratory sensor initially discussed in the previous chapter. In contrast to previous IMU-based approaches that directly use accelerometer measurements, we first estimate three-dimensional point displacements on the thoracoabdominal wall using careful estimation, isolation, and double integration of the respiratory acceleration measurements with assistance from the gyroscope. We have recently shown that our system produces estimates of respiratory displacements to an accuracy of the same order as a gold standard optical motion capture system [37]. In this chapter, we build on our displacement estimation results and demonstrate that the IMU-based displacement estimates can be used to estimate a full suite of temporal, phasic, and volumetric respiratory variables.

The remainder of this chapter is organized as follows. In Methods, we present an overview of the system hardware and the displacement estimation algorithms. We follow with an introduction to the respiratory variables estimated from the displacements. The Results section presents our preliminary experimental findings estimating a variety of respiratory variables. We conclude the chapter with a discussion and need for future work.

Methods

Hardware description

The hardware setup used in this study consists of a microcontroller, IMU, and microSD for data storage (Fig. 14). The IMU is a TDK Invensense ICM-42605 MEMS IMU containing a 3-axis accelerometer and 3-axis gyroscope with on-chip 16-bit analog to digital converters (ADCs). This particular IMU was chosen because of its good noise characteristics, and the gyroscope has a relatively low selectable full-scale range, needed to measure the rotation rates of the thoracoabdominal wall (caused by tilting of the thorax). Both the accelerometer and gyroscope are set to a 1 kHz sampling rate.

The system is powered by a rechargeable 750 mAh lithium polymer battery. A snap-together housing contains the electronics and is attached to body-worn straps using hook-and-loop closure or biocompatible tape.

The system can easily be made fully wireless and packaged in the size of a typical fitness tracker, as shown in our previous work using a wearable Bluetooth version [38]. For this study,

the system was synchronized via coaxial cable with an optical motion capture system to capture ground-truth reference data, requiring a larger housing to accommodate optical markers.

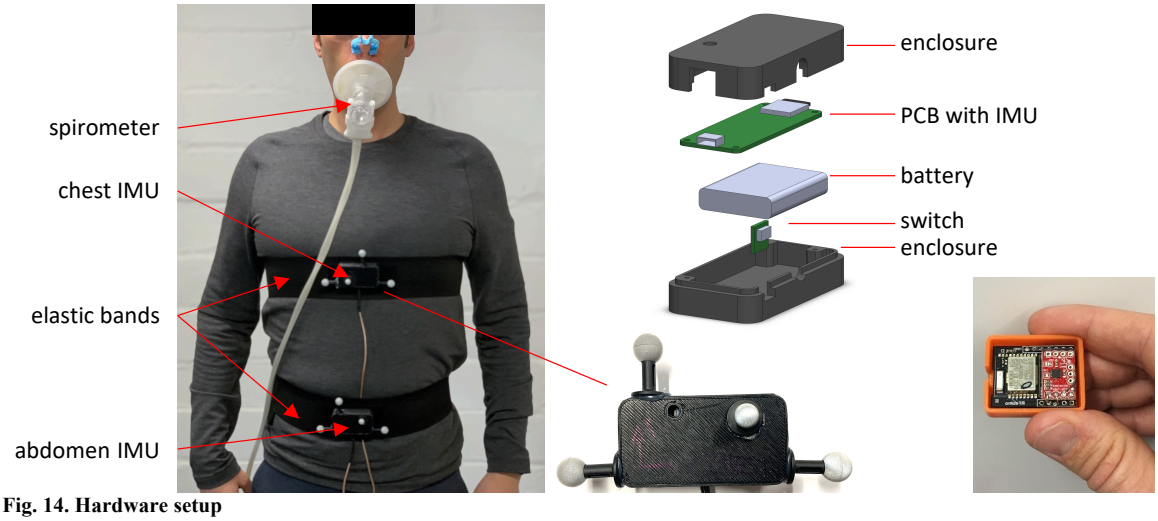

Fig. 14. Hardware setup
Image of hardware setup using two IMUs worn on chest and abdomen, and a spirometer for reference measurement. Bottom right image shows a Bluetooth version of our device and the relative size of packaging when optical reference is not needed.

Algorithm overview

Fundamentally, the objective of the algorithm is double integration of corrected acceleration measurements to obtain three-dimensional respiratory displacements. The corrections to the accelerometer's signals must compensate for bias and gravity components which have a periodic variation with back-and-forth tilting of the thorax during breathing. From the obtained thoracoabdominal displacements, temporal, phasic, and volumetric respiratory variables can be estimated. We first describe the displacement estimation algorithm and then address specific respiratory variables.

Estimation of thoracoabdominal displacements

A full treatment of the displacement estimation algorithm may be found in Chapter 2 and related publication [37], but an overview is presented again here in the context of the estimation of the temporal, phasic, and volumetric variables of interest in this chapter. Estimation of three-dimensional point displacements on the surface of the thoracoabdominal wall begin with acceleration measurements $\boldsymbol{y}_{acc}(t) \in \mathbb{R}^3$, modeled as:

$$\boldsymbol{y}_{acc}(t) = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \boldsymbol{a}_{lin}(t) - \boldsymbol{g}_S(t) + \boldsymbol{b}(t) + \boldsymbol{v}(t), \tag{32}$$

where $\boldsymbol{a}_{lin}(t)$ are the quasi-periodic accelerations of respiration we seek to isolate and double integrate, $\boldsymbol{g}_S(t)$ is gravity in the IMU frame of reference, $\boldsymbol{b}(t)$ is a bias term, and $\boldsymbol{v}(t)$ is additive noise. The bias term can be estimated offline, but even with careful calibration it is impossible to entirely remove and causes runaway drift when integrated. The bias term can be assumed to be slowly time varying at frequencies much lower than the fundamental frequencies encountered in respiration and can be removed with high pass filtering.

The term $\boldsymbol{g}_S(t)$ is the gravity vector in the IMU reference frame. If the sensor undergoes a rotation relative to gravity, then $\boldsymbol{g}_S(t)$ is a time varying term except when the axis of rotation is perfectly aligned with gravity. Even during static posture (e.g. sitting, supine, or standing) the thoracoabdominal wall undergoes small rotations on the order of 0.03 rad (2°) primarily in the sagittal plane and about the body mediolateral axis. As these rotations occur during each breath, $\boldsymbol{g}_S(t)$ and $\boldsymbol{a}_{lin}(t)$ have similar frequency spectra and cannot easily be separated. However, we utilize the rotation rates measured by the gyroscope to directly estimate $\boldsymbol{g}_S(t)$ using a fixed-interval Kalman Smoother. Once the gravity term is estimated by the smoother, it is removed from the acceleration measurement, and the residual is double integrated for respiratory displacement estimation. A reader more interested in the estimation of the respiratory variables from the thoracoabdominal displacements (rather than the estimation of the displacements themselves) can skip to the next sub-section of the paper.

In general, the fixed-interval Kalman Smoother provides better state estimates than a Kalman Filter as it uses all data points in the interval to estimate the state at every point in time. The smoother cannot be used in real-time, but for medical diagnostic applications the interest is in analyzing long-term trends using snapshots of data. The Kalman Smoother employed here uses two identical Kalman Filters that estimate the state in opposite time directions. The forward filter is initialized at the beginning of the time interval and operates in the forward time direction, while the backward filter is initialized at the end of the time interval and propagates in the backward time direction.

A fundamental issue with estimation of gravity is that it is norm constrained: it has a 1 g fixed magnitude. If two elements of $\boldsymbol{g}_S$ are known, the third can be determined by enforcing the magnitude constraint. This constraint causes issues in filtering that can be seen by considering the standard Kalman measurement update step:

$$\hat{\boldsymbol{x}}_k^+ = \hat{\boldsymbol{x}}_k^- + K(\boldsymbol{y}_k - \hat{\boldsymbol{y}}_k) \tag{33}$$

Even if the prior estimate has a 1 g magnitude, there is nothing special about the update equation that would guarantee the posterior estimate retains the magnitude of the prior. One approach to accommodating the constraint is renormalization of the estimate to 1 g every time step. Alternatively, an error-state Kalman Filter can directly incorporate the norm constraint (see Markley and Crassidis [5] for an introduction). In the error-state framework, we define an error rotation $R_{S\hat{S}}$ that rotates the estimated gravity vector, $\hat{\boldsymbol{g}}_S$, to the true gravity vector, $\boldsymbol{g}_S$, and we parameterize the error rotation with an error state vector, $\boldsymbol{\delta v} \in \mathbb{R}^3$ as follows:

$$\boldsymbol{g}_S = R_{S\hat{S}}(\boldsymbol{\delta v})\hat{\boldsymbol{g}}_S \tag{34}$$

$$R_{S\hat{S}}(\boldsymbol{\delta v}) \equiv e^{-[\boldsymbol{\delta v}]_\times} \approx I_3 - [\boldsymbol{\delta v}]_\times \tag{35}$$

The filter can now be restructured using $\boldsymbol{\delta v}$ as the state, and thus the covariance P models the uncertainty of $\boldsymbol{\delta v}$. We now find the kinematic differential equation of $\boldsymbol{\delta v}$ by noting the true kinematic differential equation for the gravity vector is given by:

$$\dot{\boldsymbol{g}}_S(t) = -[\boldsymbol{\omega}_S(t)]_\times \boldsymbol{g}_S(t) \tag{36}$$

The term $\boldsymbol{\omega}_S(t) \in \mathbb{R}^3$ is the true angular velocity of the body, and $[\boldsymbol{\omega}_S(t)]_\times$ is the skew symmetric matrix defined as:

$$[\boldsymbol{a}]_\times \equiv \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \tag{37}$$

In reality, we do not have access to the true angular velocity of the body but a noisy measurement from the gyroscope, modeled after calibration to remove bias as:

$$\boldsymbol{y}_\omega(t) = \boldsymbol{\omega}_S(t) + \boldsymbol{w}(t) \tag{38}$$

The gyroscope measurement is used to propagate the state estimate of the gravity vector:

$$\dot{\hat{\boldsymbol{g}}}_S(t) = -[\boldsymbol{y}_\omega(t)]_\times \hat{\boldsymbol{g}}_S(t) \tag{39}$$

By taking the time derivative of (34), substituting the true and estimated kinematic differential equations, (36) and (39) respectively, and utilizing the first order approximation in (35), the kinematic differential equation of the error vector $\boldsymbol{\delta v}$ is given by:

$$\boldsymbol{\delta\dot{v}} = -[\boldsymbol{y}_\omega]_\times \boldsymbol{\delta v} - \boldsymbol{w} \tag{40}$$

Equation (40) is used to propagate the state covariance. As will be shown below, $\widehat{\boldsymbol{\delta v}}$ is reset to 0 after each measurement update, and thus $\widehat{\boldsymbol{\delta v}}^{-}(k) = 0$ after the prediction step.

For the Kalman Filter measurement model, we note that the linear accelerations of interest are at least two orders of magnitude smaller than the gravity vector, so for the purposes of estimating gravity, the measurement model in terms of the error state is to a first order approximation:

$$\begin{gathered}\boldsymbol{y}_{acc} = -R_{S\hat{S}}(\boldsymbol{\delta v})\hat{\boldsymbol{g}}_S + \boldsymbol{v} \approx -\hat{\boldsymbol{g}}_S - [\hat{\boldsymbol{g}}_S]_\times \boldsymbol{\delta v} + \boldsymbol{v}, \\ \boldsymbol{v} \sim \mathcal{N}(0, \sigma_{acc}^2 I_3)\end{gathered} \tag{41}$$

The key benefit of the error state filter is that the measurement update produces an estimate of the error vector:

$$\widehat{\boldsymbol{\delta v}}^{+} = \widehat{\boldsymbol{\delta v}}^{-} + K\left(\boldsymbol{y}_{acc} + e^{-[\widehat{\boldsymbol{\delta v}}^{-}]_\times}\hat{\boldsymbol{g}}_S^{-}\right) \tag{42}$$

The error vector has no norm constraint, as the parameterization given by (35) produces a valid rotation matrix for any $\boldsymbol{\delta v} \in \mathbb{R}^3$, so the standard Kalman additive update in (42) is valid. Once the posterior estimate of the error, $\widehat{\boldsymbol{\delta v}}^{+}$, is found, it needs be incorporated into the gravity estimate using (35):

$$\hat{\boldsymbol{g}}_S^{+} = e^{-[\widehat{\boldsymbol{\delta v}}^{+}]_\times}\hat{\boldsymbol{g}}_S^{-} \tag{43}$$

By pushing the estimate of the error to the global estimate $\hat{\boldsymbol{g}}_S^{+}$, the error vector is reset to 0. According to Markley [29], the reset operation has little practical impact on the covariance.

For a fixed time interval, one gravity estimation filter is run in the forward time direction, and one in the backward time direction. These forward and backward estimates at each time point in the interval may then be combined by the smoother. The optimal smoother covariance, P^s, can be shown to be [30]:

$$P^s = \left(P^{f\,-1} + P^{b\,-1}\right)^{-1}, \tag{44}$$

where P^f and P^b are the forward and backward filter covariances, respectively. The optimal smoother estimate is then given as:

$$\hat{\boldsymbol{x}}_k^S = P^s \left({P_{k|k}^f}^{-1} \hat{\boldsymbol{x}}_{k|k}^f + {P_{k|k+1}^b}^{-1} \hat{\boldsymbol{x}}_{k|k+1}^b \right) \tag{45}$$

However, we again face the norm constraint issue, as the additive nature of mixing the estimates has no norm preservation guarantee. As in the error-state filter design, we utilize an error vector that rotates the forward estimate to the backward estimate, and we apply a portion of that rotation based on the relative covariances of the forward and backward estimates. A related method has been presented in literature in the context of smoothing unit-norm quaternions [39]. Here, we define an error vector using the cross product of the forward and backward estimates:

$$\boldsymbol{\Delta g} = \boldsymbol{g}^f \times \boldsymbol{g}^b \tag{46}$$

A rotation of θ, where $\sin(\theta) = \|\boldsymbol{\Delta g}\|_2$, about in axis pointing in the direction of $\boldsymbol{\Delta g}$ will rotate $\boldsymbol{g}^f$ to $\boldsymbol{g}^b$. To determine the smoother correction, $\boldsymbol{\Delta g}$ is weighted according to the covariances:

$$\boldsymbol{\delta g} = P^s {P^b}^{-1} \Delta \boldsymbol{g}, \tag{47}$$

Using Rodrigues' rotation formula , the smoother correction rotation is then applied to the forward estimate [32]:

$$\hat{\boldsymbol{g}}^s = \left(I_3 + [\boldsymbol{\delta g}]_\times + \sqrt{1 - \|\boldsymbol{\delta g}\|_2^2} \left[\frac{\boldsymbol{\delta g}}{\|\boldsymbol{\delta g}\|} \right]_\times^2 \right) \hat{\boldsymbol{g}}^f \tag{48}$$

Once gravity is estimated at each time step, it is subtracted from the acceleration measurement, and the residual is double integrated and high-pass filtered to obtain displacement. We have found a 6th-order Butterworth filter with a 0.07 Hz cutoff frequency provides good drift-free displacement estimates when compared to a gold standard optical motion capture system. Forward-backward filtering (Matlab "filtfilt") produces a displacement estimate with zero phase lag.

Estimation of temporal respiratory variables.

After displacement estimation, several respiratory variables may be extracted from the displacement time series, including clinically relevant temporal components of the respiratory cycle. As an example, Brack and colleagues showed that the variability in inspiratory (t_I) and expiratory (t_E) time is dependent on resistive loading of the airway [40]. The total cycle time (t_{Tot}) along with t_I and t_E are defined as the time intervals between local extrema in the respiratory volume time series. Specifically, t_I is the time interval between local extrema during inspiratory flow of a breath cycle, t_E is the time interval between local extrema during the

expiratory flow of a breath cycle, and t_{Tot} is the time interval from the start of one inspiratory cycle to the end of the following expiratory cycle [17].

Motamedi-Fakhr et al. demonstrated that the time intervals between extrema in the thoracoabdominal displacement curve measured using SLP agree with the estimates of t_I, t_E, and t_{Tot} measured with pneumotachography to within 0.3 s [17]. Similarly, we estimate these parameters using the IMU-based displacement estimates.

Peak detection was accomplished using the Matlab "findpeaks" function, which locates local extrema of a time series. We obtain best results by first low-pass filtering the anteroposterior displacement estimates to remove high frequency transients using a 6th order Butterworth filter with a cutoff frequency of 5 Hz. The displacements were forward-backward filtered to prevent phase distortion. A breath was considered for further analysis if two neighboring inhalations contained a single exhalation minimum in between in both the spirometer and IMU data. Otherwise, the breath was ignored, typically due to a transient breath interruption.

Detection of thoracoabdominal asynchrony

In normal breathing, the displacements of the rib cage and abdomen are generally in phase, meaning outward and inward motions occur in near-synchrony. The presence of relative phase lag between the two compartments is referred to as thoracoabdominal asynchrony (TAA). Paradoxical breathing is the extreme case of TAA and is characterized by compartmental displacements that are fully out of phase during respiration [41]. The presence of TAA and/or paradoxical breathing is associated with respiratory distress, as in COPD [42].

To detect and estimate TAA, we utilize the cross correlation extracted from a moving window of the anteroposterior displacements. Cross correlation has previously been shown to be the best quantitative estimator of TAA phase angle in the context of experimental RIP [43]. Defining the window size to be i samples long, then the window of anteroposterior displacements from IMU p at time step k is:

$$\boldsymbol{x}_p(k) = [x_k(k) \quad \dots \quad x_k(k+i-1)]^T \tag{49}$$

Since the abdominal and chest displacement magnitudes may not be in the same range, we first normalize the displacements such that $\boldsymbol{x}_p$ is zero mean and $\boldsymbol{x}_p^T\boldsymbol{x}_p = 1$. The estimate of the cross correlation of IMUs p and q is then given by:

$$\hat{R}_{pq}(m) = \boldsymbol{x}_p(k+m)^T \boldsymbol{x}_q(k) \tag{50}$$

For simple detection of paradoxical breathing, only $\hat{R}_{pq}(0)$ $(m = 0)$ needs to be calculated, as in-phase displacements have positive cross correlation at zero lag, and out-of-phase displacements have negative cross correlation at zero lag. Thus a simple threshold may be used to detect paradoxical breathing using $\hat{R}_{pq}(0)$.

For quantitative estimation of the phase angle, $\hat{R}_{pq}$ should be calculated across a range of delays, m. The delay m_{peak} corresponding to the peak cross correlation is then an estimate of the relative lag. The phase angle can then be estimated using the respiratory rate:

$$\hat{\phi} = 2\pi m_{peak} f_0 T \tag{51}$$

where f_0 is the respiratory rate in Hz, and T is the IMU sample period in seconds, and $\hat{\phi}$ is in radians.

Asynchrony is also commonly presented using Lissajous curves, which plot displacements of the rib cage on the vertical axis against corresponding abdominal displacements. For normal in-phase breathing, the Lissajous curve appears close to a positive slope line segment. For paradoxical breathing, the curve appears as a negative slope line segment. For intermediate phase angles, the curve appears as an ellipse. As demonstrated by Prisk et al., Lissajous curves are best used qualitatively, as cross correlation techniques are less sensitive to noise and non-sinusoidal breathing patterns [43]. Example simulations of normal and paradoxical breathing alongside corresponding cross correlation and Lissajous plots are shown in Fig. 15.

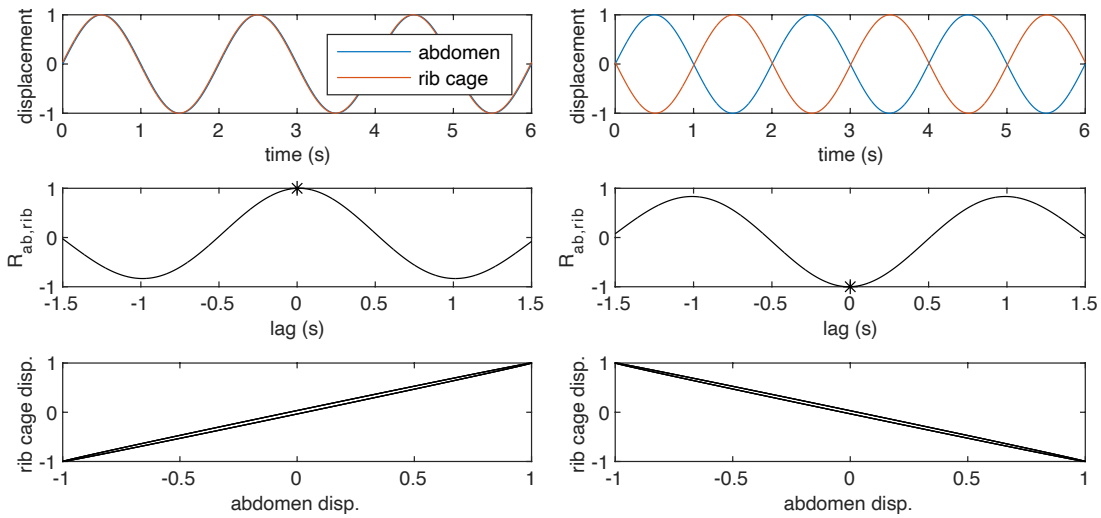

Fig. 15. Paradoxical breathing simulation
Simulation showing normal in-phase breathing (left) compared to paradoxical breathing (right). The simulated phase angle is 0.03 rad (2°)for normal and 3.1 rad (178°) for paradoxical breathing. Top row shows simulated displacements of the rib cage and abdomen. Middle row shows the cross correlation, and an asterisk shows the cross correlation at zero lag is positive for normal breathing and negative for paradoxical breathing. Bottom row shows the corresponding Lissajous curves. Because the signals are not perfectly in or out of phase, the Lissajous curve has a slight elliptical shape.

Estimation of volumetric respiratory variables

The functional relationship between thoracoabdominal displacements and respiratory volume has been the subject of much study since the 1960s [15], with techniques ranging from simple linear models to nonlinear neural networks [11]. We start with a linear two-compartment model which estimates the respiratory volume, $v(t)$, from the anteroposterior displacements of the chest and abdomen, $x_{chest,AP}(t)$ and $x_{ab,AP}(t)$:

$$v(t) = k_1 x_{chest,AP}(t) + k_2 x_{ab,AP}(t) + k_3 \tag{52}$$

The three parameters k_1, k_2, and k_3 can be estimated using training data set. From the two compartment model, we consider adding additional regressors, including the inferosuperior and mediolateral displacements, higher order terms such as peak three-dimensional distance (root sum square of the displacements), and temporal measures to capture frequency dependence in the model. The two-compartment model is useful as a baseline common in literature.

As respiratory volume analysis is most concerned with volume peaks, we fit a variety of models by matching peaks in IMU-based displacements to peaks in volume data measured with a clinical spirometer. Estimating peak volume is more practical than estimating the entire volume time series, as the spirometer and IMUs operate at different sampling rates without precise synchronization. We estimate three volume-based measures for each breath: peak inspiratory

volume (V_I), peak expiratory volume (V_E), and tidal volume (V_T), taken as the difference of V_I and V_E.

Results

Temporal respiratory variable estimation

Here, we consider the capability of the IMU-based system to estimate inspiratory, expiratory, and total cycle times (t_I, t_E, t_{Tot}, respectively) by using a set of preliminary experiments on a single subject containing an hour of supine breathing with respiratory rates ranging from 0.2 to 0.5 Hz. Overall, 980 breaths were used in the analysis. Respiration was measured using a clinical spirometer as ground-truth reference, and IMUs were attached to the abdomen and chest as described previously. The IMUs were visually oriented such that the +X sensor axis aligned with body left lateral, the +Y axis aligned with body superior, and the +Z axis with body anterior. The chest IMU was positioned over the xiphoid process, and the abdomen IMU was positioned in line with the umbilicus.

The variables were estimated in the IMU data in three ways: using the chest IMU, using the abdomen IMU, and taking the mean of the abdomen and chest t_I, t_E, and t_{Tot} estimates. Table 2 summarizes the root mean square error (RMSE) for the three temporal variables. The RMSE is also shown normalized (NRMSE) by the range of the spirometer t_I, t_E, and t_{Tot} data. For t_I and t_E, the best estimates were obtained by taking the mean of the chest and abdomen time estimates (NRMSE 0.0491 and 0.0528, respectively). In the case of t_{Tot}, the best estimates were obtained by using only the abdomen time estimates (NRMSE 0.013). Plots of t_{Tot} and t_I estimated by taking the mean of the chest and abdomen IMUs versus spirometer values are shown in Fig. 16.

TABLE 2: INSPIRATORY, EXPIRATORY AND TOTAL CYCLE TIMES ESTIMATED WITH IMU

	Chest IMU only		Abdomen IMU only		Chest and Abdomen IMUs	
	RMSE (s)	NRMSE*	RMSE (s)	NRMSE*	RMSE (s)	NRMSE*
Inspiratory Time	0.17	0.068	0.17	0.069	0.12	0.049
Expiratory Time	0.17	0.075	0.17	0.072	0.12	0.053
Total Cycle Time	0.16	0.042	0.05	0.013	0.09	0.022

*NRMSE found by normalizing the RMSE by the range in spirometer TE and TCT, respectively

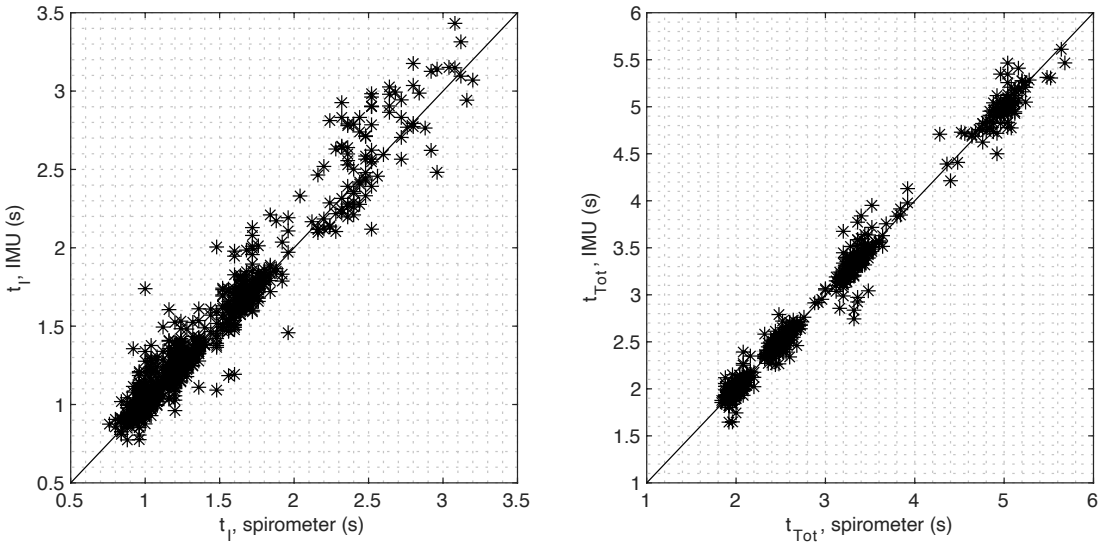

Fig. 16. Respiratory temporal variable estimation
Experimental results for estimation of t_I and t_{Tot} estimated with IMU vs spirometer.

Thoracoabdominal asynchrony detection

In previous published research studies, TAA has been induced using inspiratory resistance loading [44]. Here, we demonstrate the IMU-based respiratory sensor can detect normal breathing versus paradoxical breathing. To mimic the motions of paradoxical breathing, we measured thoracoabdominal motion while moving the abdomen in and out against a closed glottis. As this motion is isovolumetric, the chest and abdomen move paradoxically.

The experiment contains five minutes of supine breathing at 0.4 Hz with two alternating breathing patterns. The first 20 second segment contains normal breathing, while the second segment contains the closed glottis motion at the same 0.4 Hz rate. This pattern was repeated for the experiment duration. Thus, we expect to see nearly synchronous thoracoabdominal displacements during the normal breathing segments and asynchronous thoracoabdominal displacements during the breath holding segments. An example section of the experiment is shown in Fig. 17. In the displacement plot, the chest and abdomen can be seen going in and out of phase every 20 seconds. This change is detected by the near step change in cross correlation, going from +1 for normal breathing to -1 for asynchronous displacements.

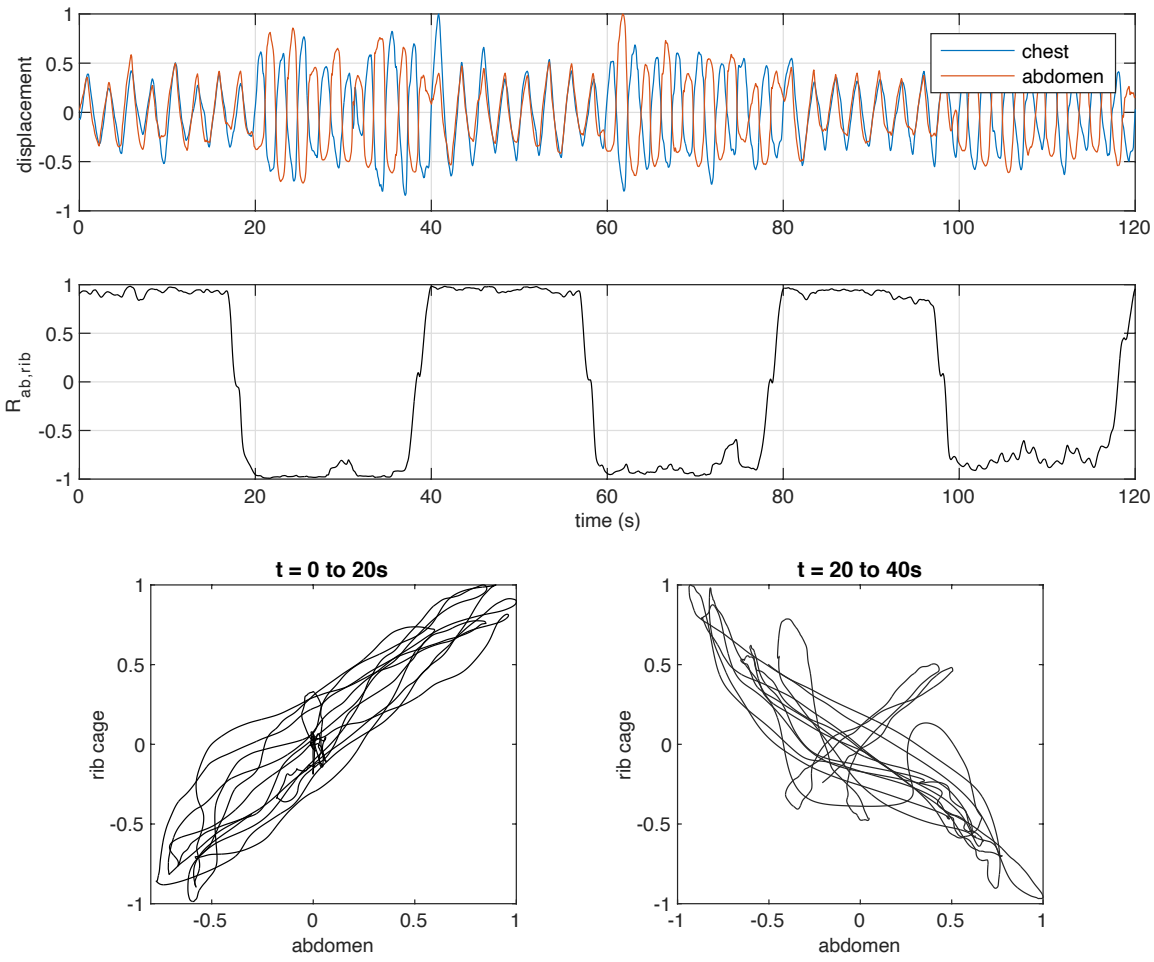

Fig. 17. Experimental detection of paradoxical thoracoabdominal displacements
Every 20 seconds, breath changes from normal (e.g. 0 to 20 s) to breath held while moving abdomen in and out (e.g. 20 to 40 s). Top plot shows anteroposterior IMU-based displacement estimates and middle plot shows the corresponding cross correlation, which changes from approximately +1 to -1 when motion changes from normal to paradoxical. Bottom two plots show the Lissajous curves for the first twenty second interval (left) and the second twenty second interval (right).

Volumetric respiratory variable estimation

To study respiratory volume estimation using IMUs, we conducted a series of identical experiments with controlled variation in respiratory rate within each experiment. The IMUs were attached to the abdomen and chest as previously noted. Each experiment consisted of an identical eight-minute supine breathing exercise containing two minutes of breathing at 0.5 Hz, followed by two minutes of breathing at 0.2 Hz, 0.4 Hz, and finally 0.3 Hz. Respiratory rate was controlled by synchronizing breathing to an auditory sound that changed rate every two minutes. Some slight variation in IMU placement occurred due to removal of the IMUs and straps in between experiments. As in the estimation of t_{Tot} and t_E above, breaths that contained transient behavior

(e.g. swallowing) were discarded. Overall, 980 breaths from over an hour of data from eight experiments were used.

After estimating thoracoabdominal displacements from the IMU data, we extracted peak displacements. Likewise, we extracted the exhalation and inhalation peaks in the spirometer data. The volume and displacement peaks were then matched by finding peaks that fell within 1 second of each other. The accuracy of the model was assessed using 5-fold cross validation. The data was randomly divided into 5 folds, and the model trained on 4 of the folds. The trained model was then used to estimate respiratory volume on the 5th validation fold, data not used for model training. This process is repeated 5 times, such that each fold is used once as the validation set. The root mean square error (RMSE) reported below is the mean RMSE of the validation folds not used for training. The RMSE is also reported as a normalized value (NRMSE) by dividing the RMSE by the range of volumes recorded by the spirometer (e.g. the range of recorded inspiratory volumes for V_I NRMSE). The models are described separately below.

Two compartment model (anteroposterior displacement only)

We first consider the baseline model using only anteroposterior displacements. Using 5-fold cross validation to fit the three function parameters k_i, the RMSE estimating V_I was 0.132 L (0.048 NRMSE, 0.94 adjusted R^2). The RMSE estimating V_E was 0.126 L (0.062 NRMSE, 0.86 adjusted R^2). The RMSE estimating V_T was 0.210 L (0.051 NRMSE, 0.92 adjusted R^2).

Inclusion of inferosuperior and mediolateral displacements

Since the IMUs estimate respiratory displacements in 3-axes, we next add the inferosuperior and mediolateral displacements to the linear model. The model in this case is:

$$v(t) = \sum_{i=1}^{3} \left(k_{chest,i} x_{chest,i}(t) + k_{ab,i} x_{ab,i}(t) \right) + k_{bias} \tag{53}$$

where i indicates the axis of the IMU (e.g. anteroposterior, inferosuperior, and mediolateral).

Using 5-fold cross validation to fit the parameters, the RMSE estimating V_I was 0.130 L (0.047 NRMSE, 0.94 adjusted R^2). The RMSE estimating V_E was 0.124 L (0.061 NRMSE, 0.86 adjusted R^2). The RMSE estimating V_T was 0.203 L (0.049 NRMSE, 0.93 adjusted R^2).

Inclusion of Total Cycle Time

If we plot the volume estimation residuals against t_{Tot} estimates, shown in left plot in Fig. 18, there is clearly a relationship between the cycle time and volume, as the estimates become biased at the extreme respiratory rates. This suggests a need to include frequency dependence to the volume estimation model. To introduce this dependence, we add t_{Tot} as a regressor:

$$v(t) = \sum_{i=1}^{3} \left(k_{chest,i} x_{chest,i}(t) + k_{ab,i} x_{ab,i}(t) \right) + k_{TCT} x_{TCT}(t) + k_{bias}, \qquad t = 1, 2, \ldots \tag{54}$$

where x_{TCT} is the estimated total cycle time of the breath.

Using 5-fold cross validation to fit the parameters, the RMSE estimating V_I was 0.118 L (0.043 NRMSE, 0.95 adjusted R^2). The RMSE estimating V_E was 0.107 L (0.052 NRMSE, 0.90 adjusted R^2). The RMSE estimating V_T was 0.170 L (0.041 NRMSE, 0.95 adjusted R^2).

Adding the t_{Tot} estimate as a regressor decreases the RMSE and increases the adjusted R^2 as compared to the baseline model. The t_{Tot} term is significant ($p < 0.05$) for all three V_I, V_E, and V_T models.

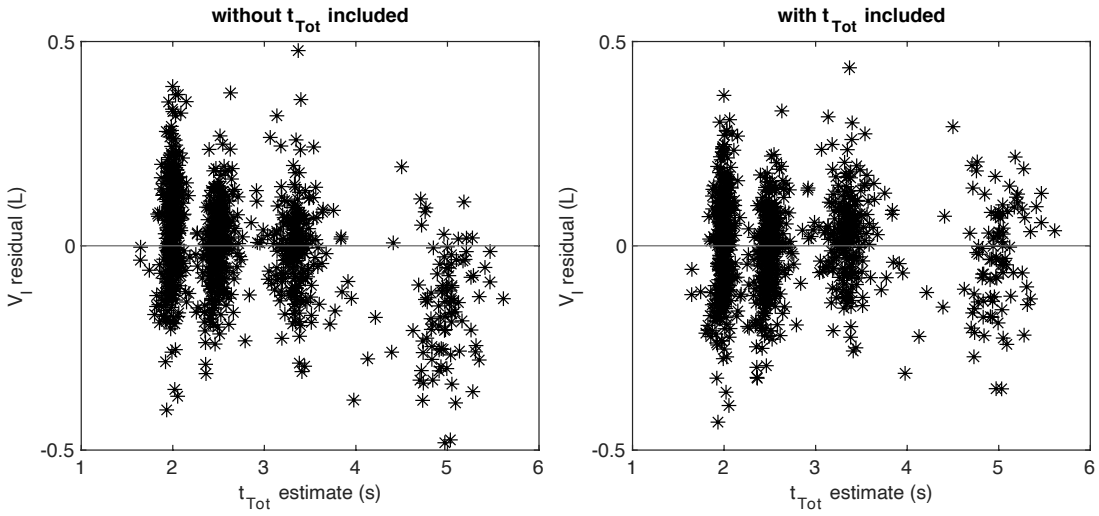

Fig. 18. Volume estimate dependence on respiratory rate
Residual of V_I estimate without (left) and with (right) t_{Tot} used as a regressor. Less frequency dependence is noted in the model that uses t_{Tot}.

Inclusion of total displacement measures

An additional consideration is whether using the peak 3-dimensional displacement (root sum square of all three axes) of each IMU improves volume estimation accuracy. We add this as a term to the model:

$$v(t) = \sum_{i=1}^{3} \left(k_{chest,i} x_{chest,i}(t) + k_{ab,i} x_{ab,i}(t) \right) + k_{RSS} RSS_{chest}(t) + k_{ab} RSS_{ab}(t) + k_{TCT} x_{TCT}(t) + k_{bias}, \qquad t = 1, 2, \ldots \tag{55}$$

Using 5-fold cross validation to fit the parameters, the RMSE estimating V_I was 0.117 L (0.042 NRMSE, 0.95 adjusted R^2). The RMSE estimating V_E was 0.106 L (0.052 NRMSE, 0.90 adjusted R^2). The RMSE estimating V_T was 0.170 L (0.041 NRMSE, 0.95 adjusted R^2).

Inclusion of higher order polynomial terms

Here, we use the three-dimensional displacements and t_{Tot} as regressors but allow higher order polynomial terms, up to sixth order. The maximum improvement in validation RMSE using higher order models is 0.003 L (V_T). As the model order increases, we risk overfitting, which is indicated when training RMSE decreases but validation RMSE remains the same or increases over simpler models, as is seen with the models above second order. A summary of all models is shown in Table 3, and plots of the volume estimates versus spirometer values for a selection of the models is shown in Fig. 19.

TABLE 3: VOLUME ESTIMATION FOR ALL MODELS

model order									V_I					V_E					V_T				
chest				abdomen					validation		training			validation		training			validation		training		
AP	ML	IS	RSS	AP	ML	IS	RSS	t_{Tot}	RMSE (L)	NRMSE	RMSE (L)	NRMSE	Adj. R^2	RMSE (L)	NRMSE	RMSE (L)	NRMSE	Adj. R^2	RMSE (L)	NRMSE	RMSE (L)	NRMSE	Adj. R^2
1				1					0.132	0.048	0.131	0.047	0.937	0.126	0.062	0.126	0.062	0.856	0.210	0.051	0.209	0.051	0.922
1	1	1		1	1	1			0.130	0.047	0.129	0.046	0.939	0.124	0.061	0.123	0.060	0.863	0.203	0.049	0.201	0.049	0.928
1	1	1		1	1	1		1	0.118	0.043	0.117	0.042	0.950	0.107	0.052	0.106	0.052	0.898	0.170	0.041	0.169	0.041	0.949
1	1	1	1	1	1	1	1	1	0.117	0.042	0.117	0.042	0.950	0.106	0.052	0.105	0.052	0.900	0.170	0.041	0.168	0.041	0.950
1				1				1	0.120	0.043	0.120	0.043	0.948	0.109	0.054	0.109	0.053	0.893	0.174	0.042	0.174	0.042	0.946
2	2	2		2	2	2		2	0.117	0.042	0.116	0.042	0.951	0.106	0.052	0.103	0.051	0.903	0.167	0.041	0.161	0.039	0.954
3	3	3		3	3	3		3	0.117	0.042	0.113	0.041	0.953	0.105	0.051	0.102	0.050	0.906	0.173	0.042	0.158	0.038	0.956
4	4	4		4	4	4		4	0.123	0.044	0.111	0.040	0.955	0.107	0.053	0.100	0.049	0.909	0.171	0.042	0.152	0.037	0.959
5	5	5		5	5	5		5	0.124	0.045	0.110	0.040	0.956	0.135	0.067	0.099	0.049	0.911	0.236	0.057	0.149	0.036	0.960
6	6	6		6	6	6		6	0.184	0.066	0.109	0.039	0.956	0.168	0.082	0.099	0.049	0.912	0.645	0.157	0.147	0.036	0.961

*Model order indicates which variables were used as regressors and to what order. For instance, a 2 indicates the first and second order terms were allowed in the model. AP = anteroposterior displacement. ML = mediolateral displacement. IS = inferosuperior displacement. RSS = root sum square displacement.

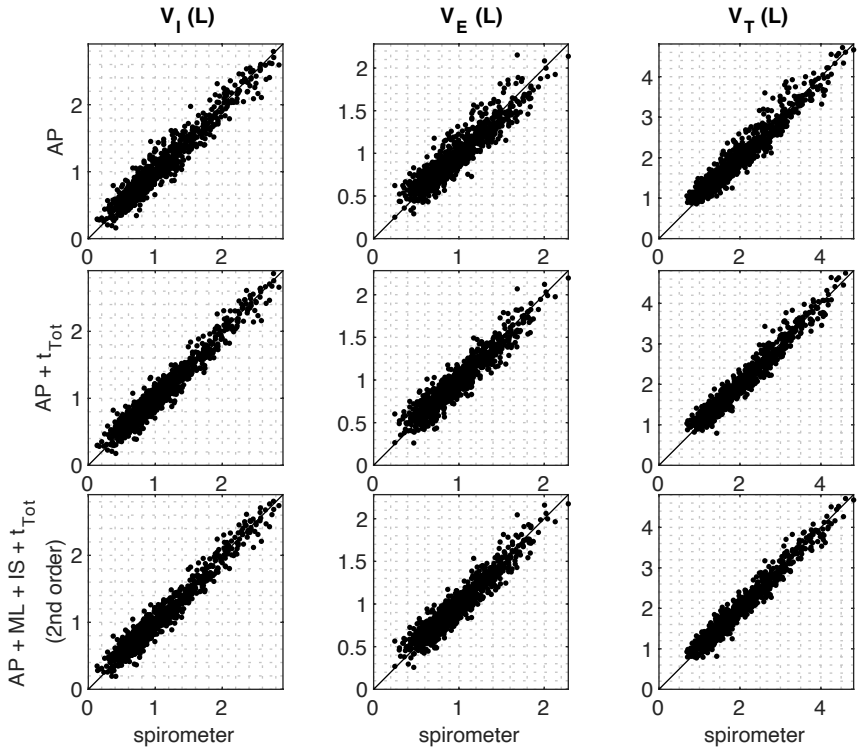

Fig. 19. Estimation of respiratory volume
IMU-based volume estimates for inhalation (left column), exhalation (middle column), and tidal volume (right column). Top model uses only anteroposterior displacements, middle row uses anteroposterior displacements and the total cycle time, and bottom row is a second order model using anteroposterior, mediolateral, and inferiosuperior displacements and the total cycle time.

Discussion

The preliminary experiments here demonstrate the capability of IMU-based respiratory monitoring and estimation of key respiratory variables. For temporal variables, inspiratory, expiratory and total cycle times estimated using peak displacements had RMSE of 0.12 seconds or better. For phase-domain respiratory variables, we demonstrated that the IMU system can detect a change from synchronous to paradoxical motions of the chest and abdomen using cross correlation.

For volume estimation, even the two-compartment anteroposterior displacement model produced estimates with NRMSE of 6.2% or better for V_I, V_E, and V_T. By adding a term to compensate for the influence of respiratory rate, the NRMSE was reduced to 5.4% or better. Further improvement was found by including the full three-dimensional displacements and higher order polynomial terms. The best model included three dimensional displacements and total cycle time up to second order. In this model, the validation NRMSE was 4.2% for V_I, 5.2% for V_E, and

4.1% for V_T. Allowing terms above second order continued to decrease the training RMSE while validation RMSE remained flat or increased, a result which indicates overfitting of the training data.

The preliminary experiments suggest the IMU is capable of accurately estimating respiratory variables in a portable package conducive to long term out-of-clinic usage. This mix of portability, accuracy, and ability to estimate temporal, phasic, and volumetric respiratory variables is not currently possible with existing technology such as spirometry, respiratory inductance plethysmography, or structured light plethysmography. The key innovation in our approach is that we estimate the three-dimensional respiratory displacements of the thoracoabdominal wall and then use the displacements to further estimate clinically relevant variables. The ability to track these variables over long durations out of the clinic has the potential to improve the diagnosis and monitoring of diseases including COVID-19, COPD, and many neuromuscular afflictions. Changes in breathing pattern can be tracked, and trends identified, thereby allowing early recognition of clinical deterioration. Conversely, as the respiratory system improves, the breathing pattern reverts predictably and gradually towards normality and can also be tracked. Our IMU sensor provides accurate estimates of breathing frequency and accurate minute volumes similar to respiratory inductance plethysmography [45], [46] with expanded capability of long-term data storage and retrieval for later review.

The clinical implications and applications of such a sensor are manifold. In the acute care setting, the decision to provide or withdraw ventilatory assistance in patients with evolving respiratory failure hinges on predictors such as $60/t_{Tot}(V_T)$ ratio which alert a clinician that a patient may not, or may, respectively, sustain spontaneous breathing [47]–[49]. Such decisions are critical in situations of ventilator shortage. Changes in thoracoabdominal movement with spontaneous breathing provide clues concerning respiratory muscle recruitment, and patterns presaging respiratory muscle failure can be recognized by artificial intelligence techniques to alarm caregivers– filling a need when limited manpower compounds equipment shortages. In the ambulatory setting, real-world respiratory monitoring becomes feasible. Optoelectronic plethysmography has already been used to study breathing kinematics in infants with spinal muscular atrophy (SMA) in the laboratory. Respiratory failure in amyotrophic lateral sclerosis (ALS) progresses over months from onset of symptoms while patients with Duchenne muscular dystrophy (DMD) generally experience a slower but inevitable onset of respiratory failure. Monitoring such deterioration is limited to measuring forced vital capacity, maximal inspiratory pressure, O_2 saturation and non-invasive PCO_2 checks. Despite the key role of respiratory failure

in morbidity and mortality associated with these disorders, there remains uncertainty concerning optimal timing to institute respiratory intervention(s). Shortcomings in current clinical strategies for predicting onset of ventilatory failure have hindered development of practice guidelines and clinical trials. Absence of a reliable prognostic model prevents clinicians from anticipating mechanically-assisted ventilation, thereby limiting the opportunity for preparing patients for shared decision making, for improving timeliness of referrals for respiratory cares and for developing clinical trial design. Objective and valid outcome measures of respiratory function are needed to test treatment efficacy in clinical trials of new therapies such as nusinersin for SMA, eteplersin in DMD, edaravone in ALS.

Chapter 4:

Detection of respiratory loading using IMUs

Introduction

In this chapter we turn our attention to measuring abnormal respiratory kinematics that result from labored breathing using inertial measurement units (IMUs). Detection of labored breathing is directly applicable to monitoring myriad diseases, ranging from neurodegenerative diseases and chronic obstructive pulmonary disease to COVID-19. It is well known that normal breathing is typically characterized by rib cage and abdominal motions that are nearly synchronous in time. However, previous research has shown that introducing inspiratory resistance in the airway results in several characteristic abnormalities to the motion. Hammer et al. found that abnormalities take the form of thoracoabdominal asynchrony of the chest and abdomen that increases with inspiratory loading [44], a trend confirmed by Tobin and colleagues who also found an increase in the breath-to-breath variability of the contribution of each compartment to the tidal volume correlated with inspiratory resistance [50]. Similarly, Brack et al. found that increasing inspiratory resistance to 3 cm H_2O/L/s reduced tidal volume variability (as measured by the standard deviation and coefficient of variation), but increasing resistance further to 6 cm H_2O/L/s increased tidal volume variability.

Here, we seek to determine if IMUs may be used to detect labored breathing resulting from airway restrictions in the form of inspiratory resistance, inspiratory threshold, and positive end-expiratory pressure (PEEP). Inspiratory resistance is a restriction of the airway on the inspiratory side of the breath. Inspiratory threshold is a threshold that must be overcome initially during inspiration but once overcome is no longer restrictive. PEEP is a form of expiratory loading in which in which the patient must exhale against positive pressure in the airway. We recently undertook a small-cohort study of five healthy patients in an IRB-approved study with the aim of detecting changes in respiratory patterns from baseline breathing to the three respiratory obstructions discussed above using our IMU-based respiratory monitor introduced in the previous two chapters.

The main contribution of this work is that this study represents the first attempt to utilize IMUs for detection of labored breathing induced through airway obstructions, to the best of our knowledge. The remainder of the chapter is organized as follows. In the Methods section, we introduce the protocol and approach utilized for detection of abnormal respiratory kinematics using the IMUs. In the Results section, we present preliminary findings from five participants in the study. Finally, the chapter is concluded with a discussion of factors which may impact successful detection of labored breathing from airway obstructions.

Methods

Protocol overview

The data presented in this chapter represent the results of an IRB-approved study conducted at an exercise physiology laboratory at the University of Minnesota Masonic Children's Hospital. Five healthy male adults above the age of 40 were recruited for the study. Table 4 summarizes the height, weight, and Body Mass Index (BMI) for each subject. Each study participant wore two IMUs- one on the chest at the xiphoid process, and one immediately superior to the naval. The plastic housings containing each IMU, microprocessor, and associated peripheral hardware were affixed directly to the skin using a double sided medical tape (3M 2477P) consisting of a thermoplastic elastomer backing film with acrylic adhesive on the device side and silicone adhesive on the skin side. Each participant wore a nose clip and was instructed to breath naturally without any coaching into a mouthpiece containing a pitot tube spirometer. Each session consisted of two identical series of experiments performed in the supine and seated position. Each individual experiment lasted at minimum of two minutes but was extended to up to four minutes until the spirometer recorded 12 breaths, and the participant was allowed to remove the nose clip and breath without the mouthpiece in between each experiment. Each complete session lasted approximately 90 minutes from start to finish. The first experiment consisted of unrestricted respiratory data as a baseline. Then, three inspiratory resistance experiments were conducted with a different resistance value for each experiment (10, 20, and 50 cm H_2O/L/s). Next, three inspiratory threshold experiments were conducted with a different threshold for each experiment (10, 20, and 30 cm H_2O). Finally, two experiments with two levels of PEEP were conducted (5 and 10 cm H_2O).

TABLE 4:
HEIGHT, WEIGHT, AND BMI FOR STUDY PARTICIPANTS

Patient	Height (cm)	Weight (kg)	BMI
A	181	88.1	26.89
B	182	94.8	28.62
C	183	89.4	26.55
D	180	112.9	34.84
E	172	97.6	32.99

Detection algorithm approach

The overall goal of the present study is to determine the viability of the IMUs to detect labored breathing in diseased patients (e.g. as in chronic obstructive pulmonary disease). Initially, the intention was to estimate respiratory displacements and subsequently volumes using the IMUs, and to then use measures related to the estimated volumes as a basis for detection of loaded breathing. However, we found that the respiratory rates encountered in the study are relatively low- with rates almost entirely below 0.25 Hz and rates as low as 0.1 Hz commonly encountered. This range of respiratory rates is particularly challenging for the IMUs. See Chapter 2 of this thesis for a discussion of the issue and Fig. 11 as an example of the degradation of displacement accuracy at the low end of respiratory rates. Thus, we will instead seek approaches to detect loaded breathing that do not directly require estimation of respiratory volumes. It is expected that real-life situations involving respiratory distress (such as in COVID-19) will involve higher respiratory rates, and the low rates encountered in this study are due to the study protocol involving healthy subjects and respiratory loading.

Referring again to Fig. 11, we note that for axes nominally aligned with gravity (e.g. anteroposterior when supine), the respiratory frequency and phase remains correct, even if the peak displacements degrade. Thus, we hypothesize that even in the cases of relatively low respiratory rates, we may be able to detect changes in, for instance, thoracoabdominal asynchrony (TAA) resulting from labored breathing. It is well known that inspiratory resistance can induce TAA, and a previous study in rhesus monkeys found a mean phase angle shift of 39° between baseline unloaded breathing and an inspiratory resistance of 50 cm H_2O/L/s [44]. The study found that respiration was not close to paradoxical until an inspiratory resistance of 1000 cm H_2O/L/s was reached, on average. And so, using rhesus monkeys as a model, we would expect somewhat subtle phase shifts at the inspiratory resistances used in our study. As discussed in detail in the previous chapter, we can monitor phase shifts both qualitatively using Lissajous curves by

plotting abdominal displacements versus thoracic displacements and quantitatively using cross correlation. Recall, the phase angle may be estimated by calculating the cross correlation $\hat{R}_{\text{ab,thorax}}$ between the abdomen and rib cage at various delays m for a portion of the recorded signal at time step k:

$$\hat{R}_{ab,thorax}(m) = \boldsymbol{x}_{ab}(k+m)^T \boldsymbol{x}_{thorax}(k) \tag{56}$$

where $\boldsymbol{x}_a(k)$ is a vector of displacement estimates starting at time step k. The lag m_{peak} corresponding to the peak cross correlation is then used to estimate of the phase angle:

$$\hat{\phi} = 2\pi m_{peak} f_0 T \tag{57}$$

where f_0 is the respiratory rate in Hz, and T is the IMU sample period in seconds, and $\hat{\phi}$ is in radians. Example Lissajous curves for phase angles ranging from 0 to 60° are shown in Fig. 20.

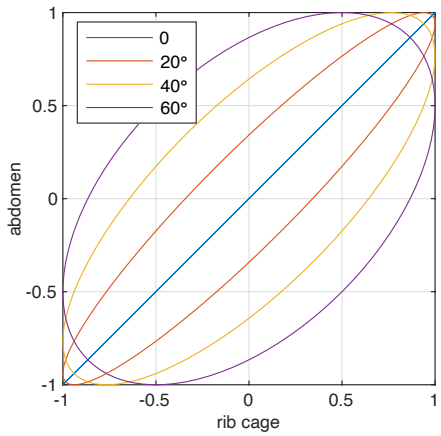

Fig. 20. Lissajous curves for phase angles ranging from 0 to 60°

Results

Overall TAA phase shift from baseline to loaded airway

The mean phase shift for all five study participants for all airway loadings is shown in Fig. 21. This figure essentially captures the overall phase shift seen on average for loaded vs. unloaded breathing for the three types of airway obstructions. A mean phase shift between 15 and

17 degrees was estimated for the inspiratory loadings, while PEEP showed essentially no phase shift. The subsections that follow will present the results for each specific loading type in detail.

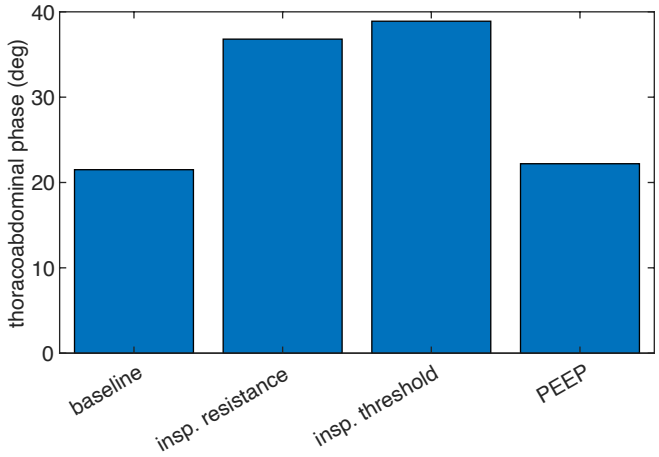

Fig. 21. Mean TAA phase shift from baseline to loaded airway
Plot of mean IMU-estimated thoracoabdominal phase for five study participants of baseline supine respiration as compared to a loaded airway for three types of loading: inspiratory resistance, inspiratory threshold, and PEEP. Each class of loading includes all gradations within that class (e.g. the inspiratory resistance bar includes the mean phase shift across all 10, 20, and 50 cm H_2O/L/s loaded experiments.

Inspiratory Resistance Loading TAA

Lissajous curves of the estimated displacements for baseline and maximum 50 cm H_2O/L/s inspiratory resistance supine experiments are shown in Fig. 22 for the five participants. Representative Lissajous curves for a single study subject for baseline and the three levels of resistance are shown in Fig. 23. Note, although the participants were instructed to lay and sit still, occasionally a participant had transient gross body movement to cough or adjust the spirometer mouthpiece. These periods of motion were noted and removed from the dataset. Qualitatively from the figures, the Lissajous curves appear less eccentric in the presence of inspiratory resistance for participants A, B, and C, an indication that TAA has increased from a roughly synchronous baseline. However, no clear trend is noted for participant E. In comparing the Lissajous curves in Fig. 22 to the simulated curves in Fig. 20, the experimental curves suggest an increase in TAA on the order of a 20 degree phase shift for participant A and B. The phase shift in TAA estimated using the cross-correlation method during the first minute of the baseline and three resistances are shown in Table 5 for all five participants. The IMU-based displacement estimates show a mean TAA phase shift above baseline for the three inspiratory resistances ranging from 9° to 24°, although the phase shift does not appear directly correlated with

resistance, as the mean phase shift decreased from a loading of 20 to 50 cm H_2O/L/s. Interestingly, one participant (D) who had the largest estimated phase angle at baseline, showed a decrease in estimated TAA phase angle with inspiratory resistance loading.

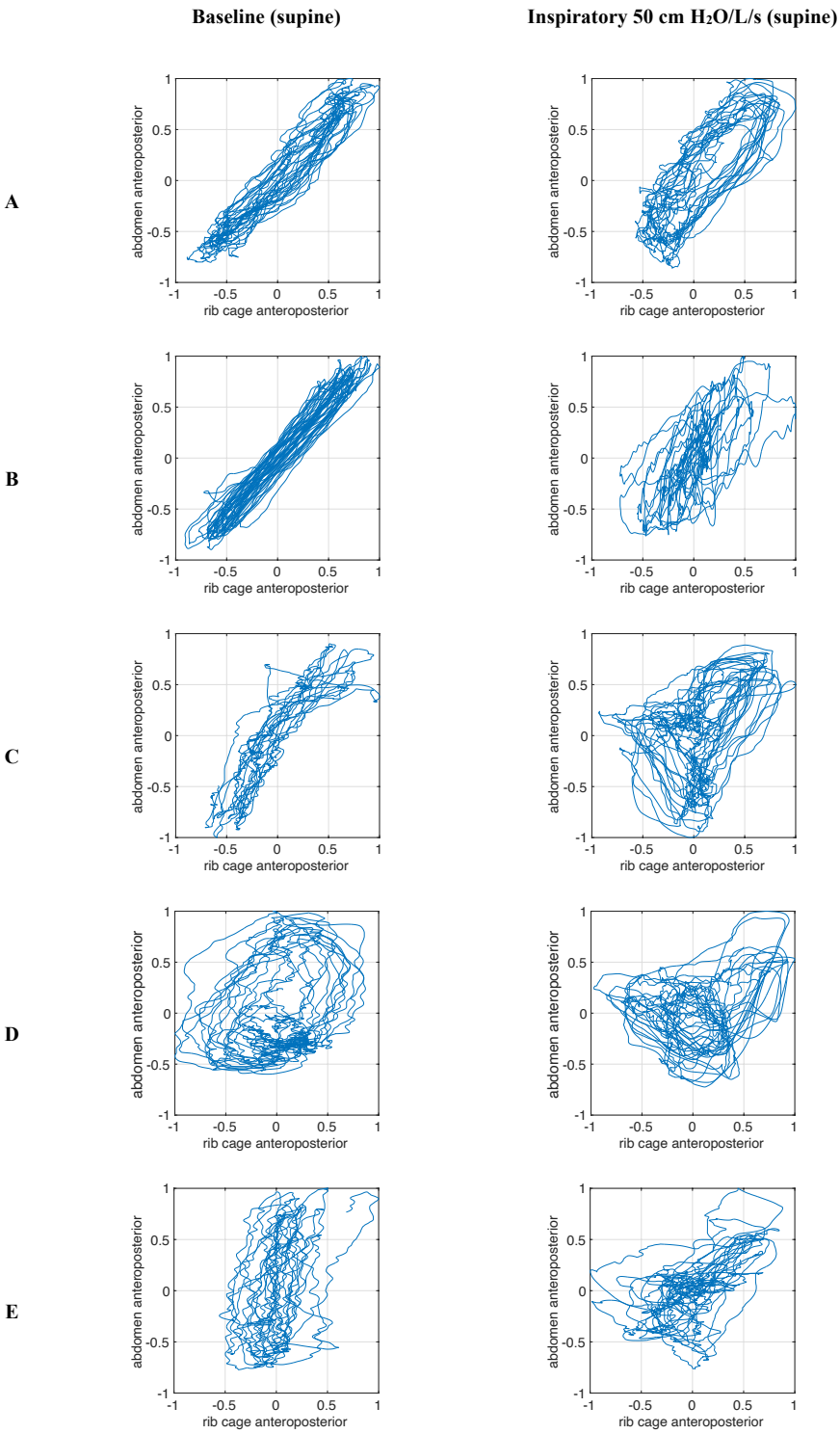

Fig. 22. Lissajous curves with inspiratory resistance

IMU-based anteroposterior Lissajous figures of baseline supine respiration compared to respiration with inspiratory loading for three participants. Displacements are normalized such that the maximum displacement is 1.

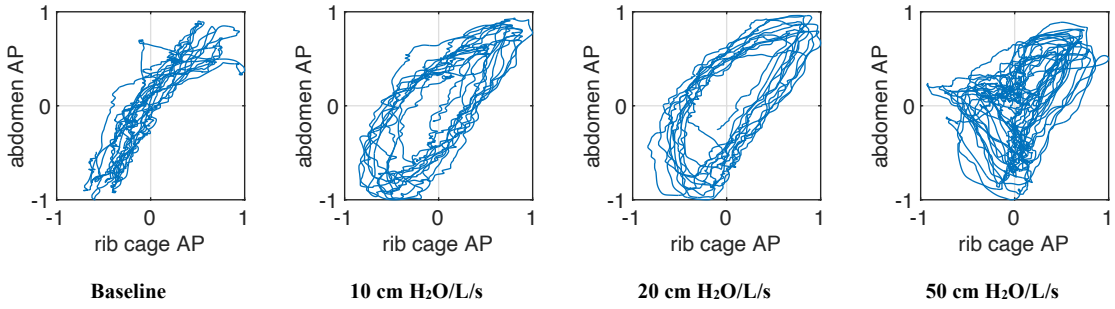

Fig. 23. Lissajous curves for subject C under inspiratory resistance
Lissajous figures of baseline supine respiration compared to respiration with inspiratory resistance loading for participant C.

TABLE 5:
ESTIMATED PHASE ANGLE FOR BASELINE VS. RESISTANCE LOADED BREATHING

Participant	Baseline	10 cm H_2O/L/s	20 cm H_2O/L/s	50 cm H_2O/L/s
A	10.9°	31.6°	67.2°	23.8°
B	2.3°	8.8°	12.7°	24.2°
C	21.7°	44.5°	64.8°	63.7°
D	44.3°	39.2°	39.4°	33.7°
E	28.4°	28.7°	42.8°	27.5°
Mean	21.5°	30.5°	45.4°	34.6°

Inspiratory Threshold Loading TAA

Lissajous curves of the estimated displacements for baseline and 30 cm inspiratory threshold supine experiments are shown in Fig. 24 for the five participants. Representative Lissajous curves for a single study subject for baseline and the three levels of inspiratory threshold are shown in Fig. 25. As in the inspiratory resistance experiments, transient periods of gross body motion were noted and removed from the dataset. Qualitatively, the Lissajous curves in Fig. 24 appear less eccentric in the presence of inspiratory thresholds, an indication that TAA has increased from the baseline. The threshold Lissajous curve for participant B is particularly interesting and suggests a change in phase over the course of the experiment. Anecdotally, participant B reported that he found the inspiratory threshold experiments particularly difficult in comparison to the other experiments. The Lissajous curves in Fig. 24 suggest an only subtle increase in TAA for participant A with a more significant increase for participant C. The phase shift in TAA estimated using the cross-correlation method during the first minute of the baseline and three threshold values are shown in Table 6 for all five participants. The IMU-based displacement estimates show a mean TAA phase shift above baseline for the three inspiratory thresholds ranging from

15° to 20°, although the phase shift does not appear directly correlated with threshold level, as the mean phase shift was maximum at a threshold of 10 cm H_2O. Except for subject D, all of the participants exhibited an increase in TAA from baseline for all of the threshold experiments. For subjects B and C, the phase angle increased with each loading increase.

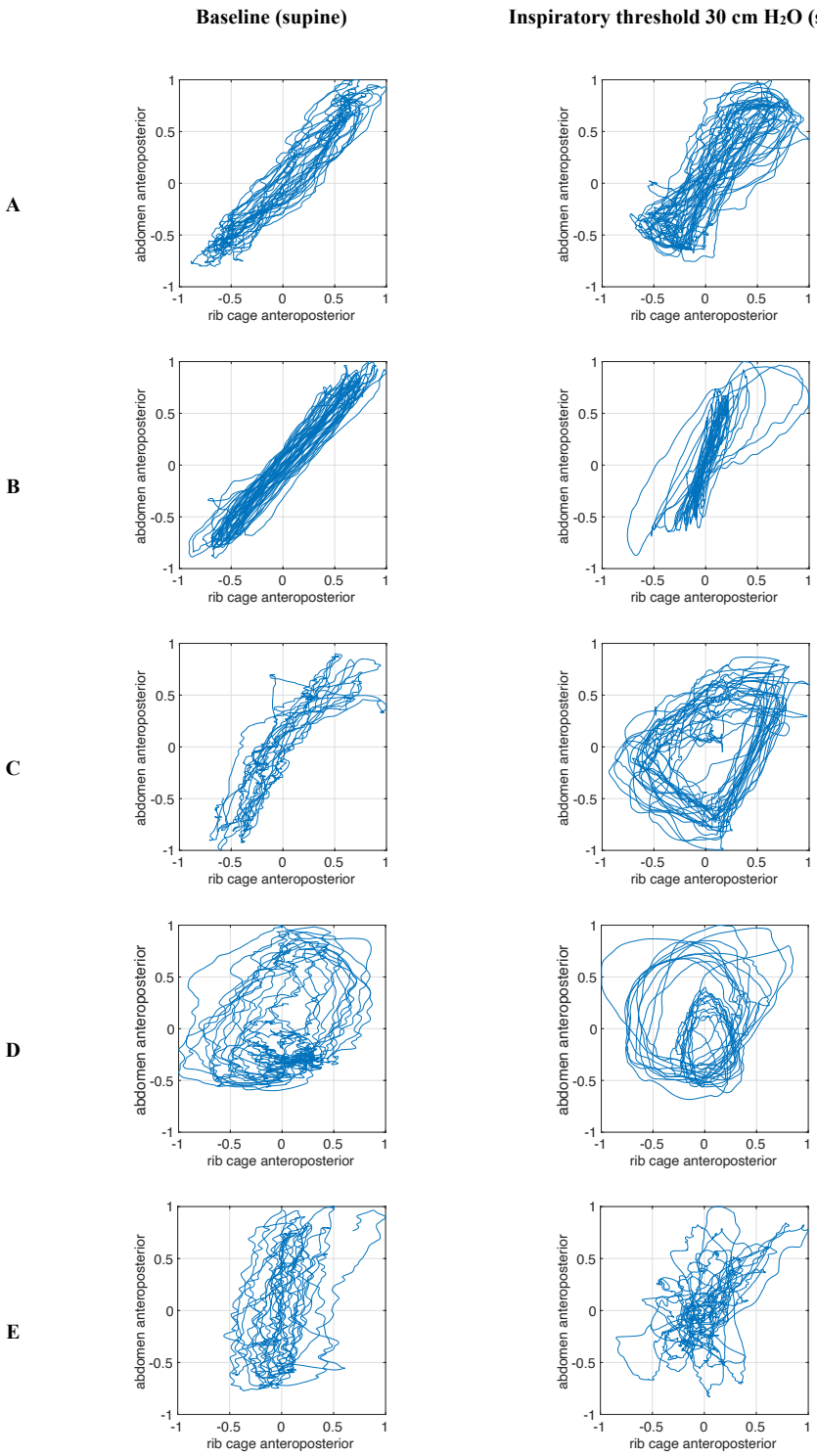

Fig. 24. Lissajous curves with inspiratory threshold
Lissajous figures of baseline supine respiration compared to respiration with inspiratory threshold loading for three patients. Displacements are normalized such that the maximum displacement is 1.

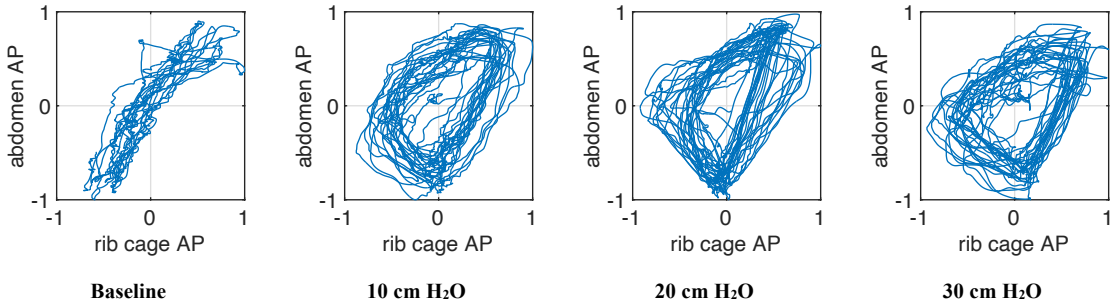

Fig. 25. Lissajous curves for subject C under inspiratory threshold
Lissajous figures of baseline supine respiration compared to respiration with inspiratory threshold loading for participant C.

TABLE 6:
ESTIMATED PHASE ANGLE FOR BASELINE VS. THRESHOLD LOADED BREATHING

Participant	Baseline	10 cm H_2O	20 cm H_2O	30 cm H_2O
A	10.9°	31.5°	29.0°	16.6°
B	2.3°	9.2°	11.1°	11.2°
C	21.7°	71.6°	76.7°	79.4°
D	44.3°	36.4°	37.2°	35.8°
E	28.4°	59.7°	30.4°	47.4°
mean	21.5°	41.7°	36.9°	38.1°

PEEP Loading TAA

Lissajous curves of the estimated displacements for baseline and 10 cm PEEP supine experiments are shown in Fig. 26 for the five participants. As in the inspiratory resistance experiments, transient periods of gross body motion were noted and removed from the dataset. Qualitatively, there are no trends in the Lissajous curves. The curve for Participant C suggests and increase in TAA under PEEP, but the curves for participant A and B do not. The phase shift in TAA estimated using the cross-correlation method during the first minute of the baseline and two PEEP values are shown in Table 7. There was essentially no change in the mean estimated TAA for the five study subjects, though there was wide individual variation.

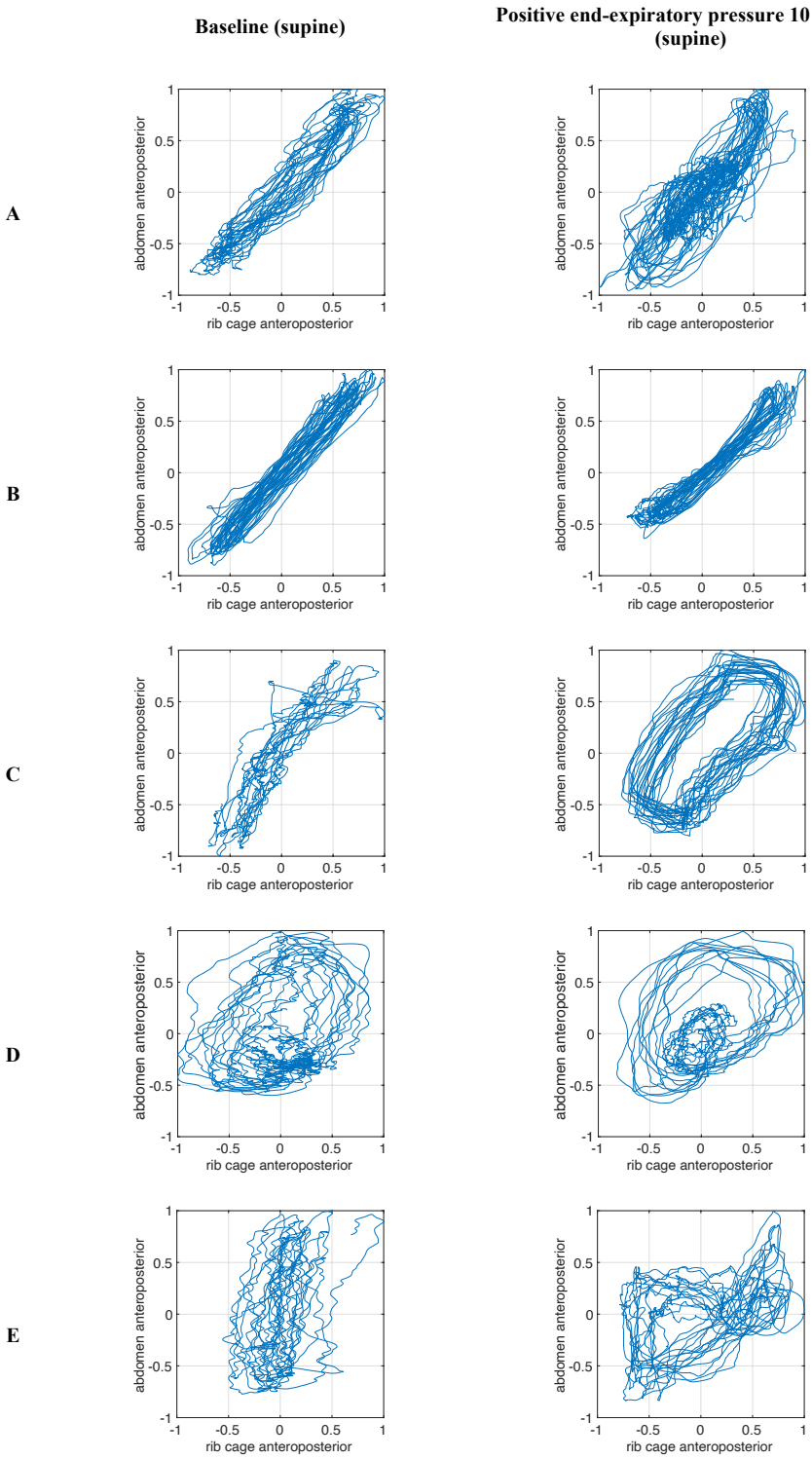

Fig. 26. Lissajous curves with PEEP

Lissajous figures of baseline supine respiration compared to respiration with positive end-expiratory pressure (PEEP) loading for three patients. Displacements are normalized such that the maximum displacement is 1.

TABLE 7:
ESTIMATED PHASE ANGLE FOR BASELINE VS. PEEP LOADED BREATHING

Patient	Baseline	5 cm H2O	10 cm H2O
A	10.9°	25.0°	0.0°
B	2.3°	2.0°	2.1°
C	21.7°	49.0°	56.4°
D	44.3°	27.7°	28.5°
E	28.4°	16.4°	15.7°
mean	21.5°	23.8°	20.5°

Variational patterns

We next turn our attention to identifying variational trends in the respiratory data, recalling from the chapter introduction that previous research has found that tidal volume variability has been found to first decrease with small inspiratory resistance and then increase with larger inspiratory resistances. We first look for such variational trends in the spirometry data and then see if similar trends can be found in the IMU peak displacement data. We will consider the abdominal anteroposterior peak displacements, which generally have the largest displacements during respiration. Table 8 reports the mean, standard deviation (SD), and coefficient of variation (CV) of the spirometry tidal volumes for baseline, 50 cm H_2O/L/s inspiratory resistance, 30 cm H_2O inspiratory threshold, and 10 cm H_2O PEEP. Table 9 reports corresponding data for the abdominal anteroposterior displacements.

No trends are seen in the spirometry data in terms of mean, SD, and CV of the tidal volumes. For instance, the mean tidal volume between baseline and 50 cm H_2O/L/s inspiratory resistance stayed close to the same for participants A and C but nearly doubled for participant B. Similarly, participant A saw nearly a doubling of tidal volume between baseline and 10 cm H_2O PEEP, while participant B saw a marginal increase and participant C a marginal decrease. Similar results were observed for SD and CV. On average across all participants, the tidal volumes increased under loaded conditions, but the CV in the tidal volume stayed nearly the same. In the IMU data, we also do not find any significant trends common to all study subject. There is an increase in mean displacement CV from baseline to loaded conditions, but most of that increase comes from subject D. If we remove subject D from the dataset (see Discussion section), then there is no change in displacement CV on average.

TABLE 8:
MEAN, STANDARD DEVIATION, AND COEFFICIENT OF VARIATION OF TIDAL VOLUME (SUPINE)

Participant	Baseline			Inspiratory 50 cm H_2O/L/s			Threshold 30 cm H_2O			PEEP 10 cm H_2O		
	Mean	SD	CV	Mean	SD	CV	Mean	SD	CV	Mean	SD	CV
A	1145	93	0.081	946	145	0.153	1298	98	0.08	2051	198	0.097
B	821	82	0.100	1660	478	0.288	1958	386	0.20	1008	140	0.139
C	1150	171	0.150	1169	165	0.141	1375	191	0.14	1090	112	0.103
D	1364	177	0.130	3552	536	0.15	2954	118	0.04	N/A	N/A	N/A
E	1042	196	0.18	1882	331	0.18	1771	143	0.08	1683	364	0.22
Mean	1104	144	0.13	1842	331	0.18	1871	187	0.11	1458	204	0.14

*units: mL, N/A = not available

TABLE 9:
MEAN, STANDARD DEVIATION, AND COEFFICIENT OF VARIATION OF ABDOMINAL ANTEROPOSTERIOR DISPLACEMENTS (SUPINE)

Participant	Baseline			Inspiratory 50 cm H_2O/L/s			Threshold 30 cm H_2O			PEEP 10 cm H_2O		
	Mean	SD	CV	Mean	SD	CV	Mean	SD	CV	Mean	SD	CV
A	9.6	1.2	0.12	5.7	0.8	0.15	8.3	1.1	0.13	8.0	1.3	0.16
B	9.2	1.3	0.14	7.2	1.5	0.21	14.6	3.4	0.24	11.6	1.3	0.11
C	5.3	0.7	0.14	5.1	0.6	0.13	6.6	0.9	0.14	7.4	0.8	0.10
D	12.3	1.3	0.11	6.2	2.3	0.37	10.4	5.7	0.55	11.8	7.5	0.64
E	9.1	1.9	0.20	8.4	1.7	0.20	10.2	3.2	0.31	5.3	1.6	0.31
Mean	9.1	1.3	0.14	6.5	1.4	0.21	10.0	2.9	0.27	8.8	2.5	0.26

*units: mm

Comparison to optical reference system

In this subsection, we compare TAA estimated with the IMUs to TAA estimated with the reference optical system (NaturalPoint Inc. optiTrack V120:Trio, see Chapter 2 for an introduction). Here, we exclude the data from subject B, as there was significant marker occlusion in the optical data due to lighting conditions and camera position.

The shift in TAA from baseline estimated with the IMU as compared to the optiTrack is shown in Fig. 27. Overall, the IMU and optiTrack show similar trends. For instance, for subject A, both the IMU and the optiTrack show an initial increase in TAA under the two lower inspiratory resistances that then decreases at 50 cm H_2O/L/S. Despite an overall similarity in TAA trends between the two systems, the actual phase angle difference between the IMU and optiTrack was as high as 40° (subject A, 20 cm H_2O/L/S inspiratory resistance). The optiTrack system confirms the apparent decrease in TAA from baseline estimated with the IMUs for subject D. The best agreement between the two systems was in subject C. A comparison of IMU-derived and optiTrack-derived Lissajous curves are shown in Fig. 28 for subject C.

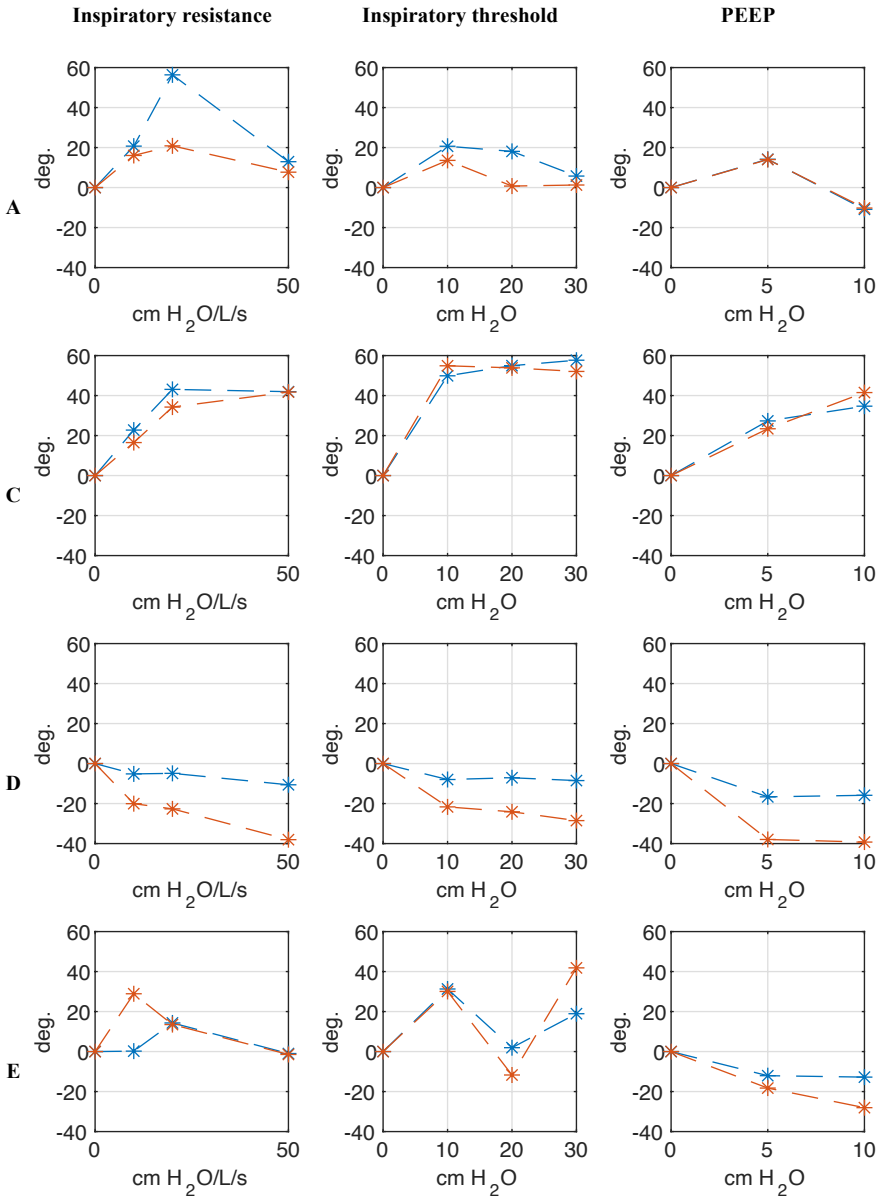

Fig. 27. TAA from baseline, IMU vs. optiTrack
Comparison of phase shift from baseline determined with IMU (blue) and optiTrack (red). A positive value on the vertical axis indicates an estimated increase in asynchrony, while a negative value indicates and estimated decrease in asynchrony.

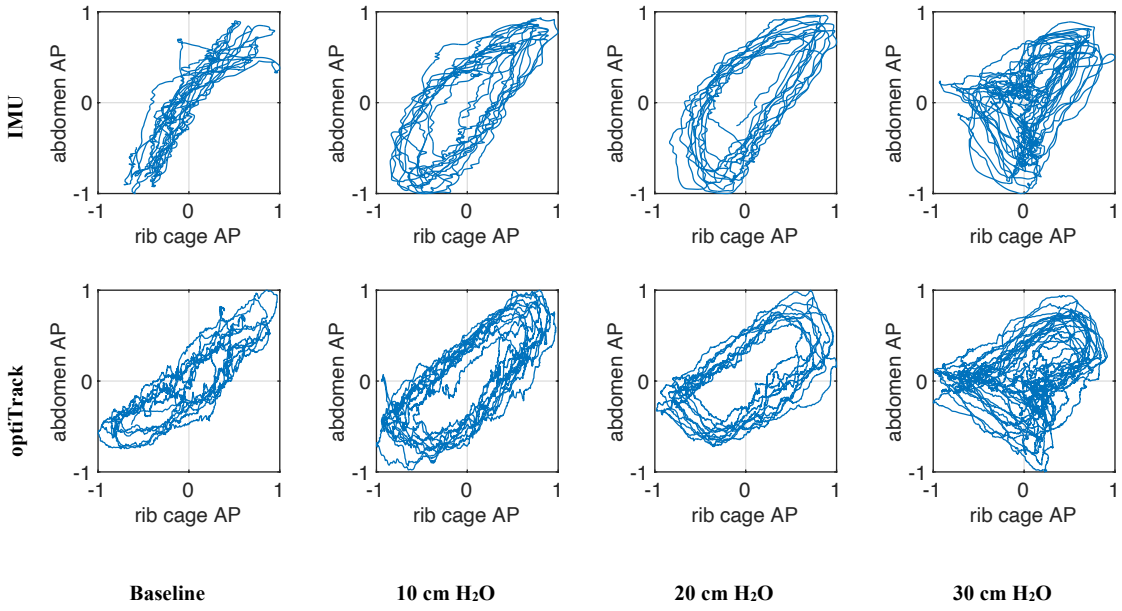

Fig. 28. IMU vs. optiTrack Lissajous curves for subject C under inspiratory resistance
Lissajous figures based on IMU data (top) compared to optiTrack data (bottom) of baseline supine respiration compared to respiration with inspiratory resistance loading for participant C.

Discussion

Although limited conclusions may be drawn due to the small number of participants in the study, the experimental results presented showed feasibility for detection of TAA resulting from an obstructed airway in some of the study participants. Even under challenging respiratory rates for the IMU (less than 0.25 Hz and as low as 0.1Hz), a mean TAA phase shift of 15 and 17 degrees was estimated for inspiratory resistance and threshold, respectively, a shift which compares favorably to previously reported research. Across all participants, no trend in the mean phase shift from PEEP loading was found, although TAA resulting from expiratory loading has not been previously reported, to the best of our knowledge.

Although we found a mean increase in thoracoabdominal phase resulting from inspiratory loading, the results are mixed when looking at specific individuals, and we consider individual factors which may impact successful use of the IMU as a respiratory sensor in this context. For instance, previous research suggests that TAA should increase as inspiratory resistance increases. Participants A, B, and C came the closest to exhibiting this trend in both inspiratory loadings, and these three study subjects had the clearest Lissajous figures. On the other hand, Participant D

exhibited an estimated baseline TAA significantly higher than the other participants and a relatively large decrease in TAA under loaded airway conditions. Participant E showed an increase in TAA of just over 14° under the 20 cm H_2O/L/s resistance but virtually no shift at the other inspiratory resistances. One possible explanation of the findings is that Participants D and E have significantly higher BMI (35 and 33, respectively) compared to the other three participants (less than 29). According to CDC classifications, Participants A, B, and C are considered overweight, while D and E are Class I obese. Given that Participant B had the clearest Lissajous figures and the clearest TAA trends while also the lowest BMI, it's conceivable that IMU displacement estimates may be poor for individuals with higher levels of adiposity. Our initial development work on the device described in the previous two chapters was completed on individuals with BMIs less than 27, and so the study results may be exhibiting a limitation of conditions for which the IMU works successfully.

On the other hand, when comparing the IMU-based and optiTrack-based TAA estimates, we found that both systems largely agreed on overall TAA trends for each participant, and so the protocol may not have been inducing the expected asynchrony in all subjects. For instance, both systems showed an increase in TAA from baseline for all loadings in subject C, and both systems agreed that subject D had peak asynchrony at baseline that then decreased with loading. It should be emphasized, though, that the actual phase angle disagreement between the systems was as large as 40°, and so the IMU may be prone to large errors at such low respiratory rates despite good tracking on average of TAA trends. The subject with the lowest BMI (C) had the lowest error between the two systems, while the highest BMI subject had the largest overall error (D), again indicating there may be a connection between body morphology and IMU performance. These results suggest that the IMU may be best limited to coarse estimates and/or large trends of asynchrony when respiratory rates are lower than 0.25 Hz. Further, given the large variation in individual trends, the results may also suggest that a more aggressive experimental protocol is needed to induce higher levels of TAA in future studies.

Incidentally, an especially strong respiratory signal was present in the abdominal gravity estimate for participant D, indicating a fairly significant rotation occurred during each of his breath cycles. In fact, the rotation alone was an indicator of respiratory loading in subject D's case, as shown in Fig. 29. Specifically, if we consider the time in the breath cycle at which peak rotation occurred during inhalation, we can see an increase from baseline during inspiratory resistance and a decrease from baseline during PEEP. Unfortunately, this pattern was not apparent in all five participants, and it's unclear how the exact position of the IMU on the

abdomen might impact the rotation signal. In several subjects, the respiratory signal could not be detected at all using rotation, so individual variation in respiratory kinematics due to adiposity may be playing a roll.

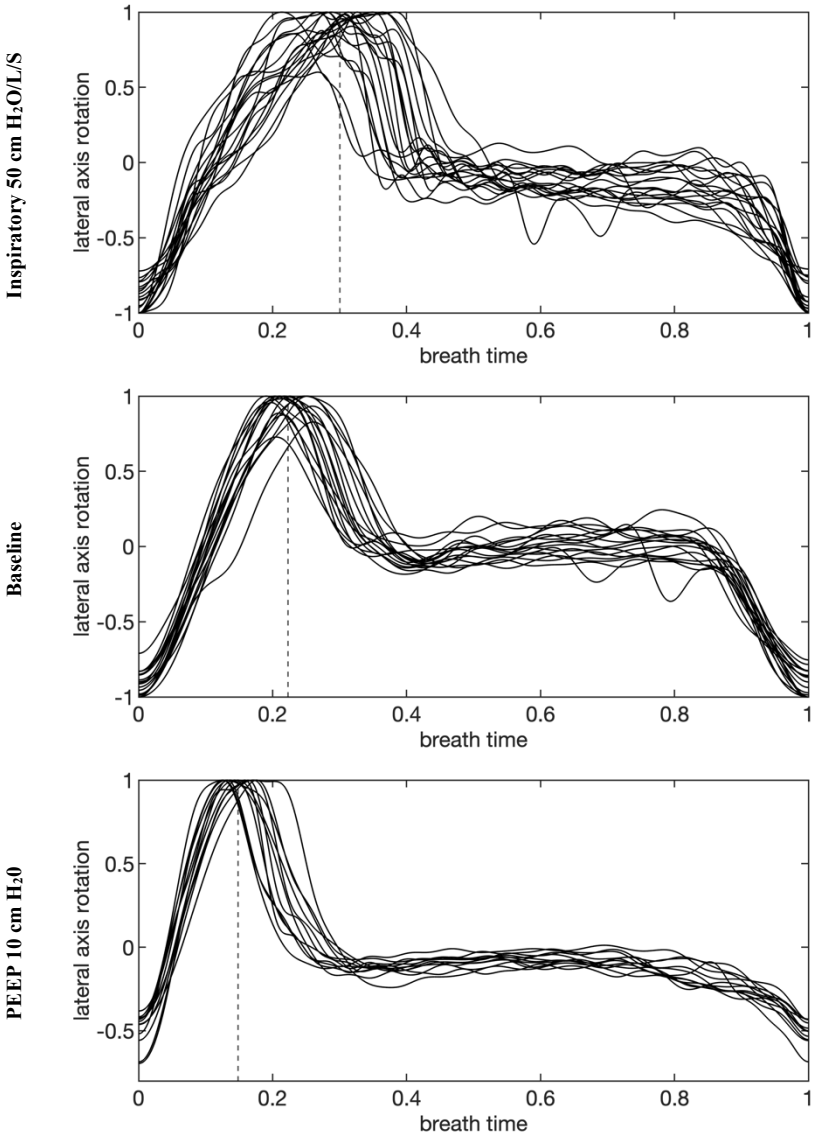

Fig. 29. Peak rotation during the breath cycle as a function of airway loading for a single participant
Three plots showing the mean time (dashed line) at which peak lateral rotation occurs during the breath cycle for participant D plotted against all breaths measured during each respective experiment. Rotation was found using the inferosuperior component of the gravity estimate found with the Kalman smoother. Both the time and rotation axes are scaled such that the breath cycle starts at 0 and ends at 1, and so that the peak rotation is scaled to 1. Plot shows a lengthening of the inspiratory side for inspiratory loading and a lengthening of the expiratory side during PEEP loading as compared to baseline.

The inconsistency of the experimental results indicates that body morphology may be a critical factor in successful use of IMUs for respiratory monitoring, in addition to the other factors

discussed previously in the thesis, namely respiratory rate and body orientation relative to gravity. A future study directly addressing the effects of adiposity is warranted.

We also examined the data for trends in breath-to-breath variability as a result of loading, as previous research has found that variability in compartmental contribution to each breath can increase under loading [50]. However, we did not find significant variational trends amongst the five participants, either in the spirometer or the IMU data. This result is possibly due to the small number of participants in the study and the length of time of each experiment. For instance, one previous study looking at tidal volume variability as a result of loaded breathing collected data for 1 hr at each resistance level [40]. Our study exposed the participant to two minutes of loaded breathing, albeit at significantly higher resistances than [40].

Despite mixed results, we emphasize again that the respiratory rates encountered in this study were relatively low, between 0.1 and 0.25 Hz. Our previous work has shown that the accuracy of the IMU degrades at frequencies below 0.25 Hz, and so the frequencies in this dataset are particularly challenging. Under the study protocol, the subjects were instructed to breath naturally, and so we did not explicitly prescribe higher rates of breathing that we are likely to find in actual diseased patients or young children. As discussed previously in the thesis, children and COPD patients in particular have higher respiratory rates than healthy adults, and so the IMU may be best targeted as a device for young or diseased adult patients. Nonetheless, the results demonstrated here are encouraging, particularly in study participants with low BMI. A larger study is needed before claims regarding the accuracy and reliability of estimating the phase shifts can be made, and the long term goal of the study is to be expanded to include approximately 10 male and 10 female participants.

Chapter 5:

Real-time detection of food intake using IMUs

Introduction

The rapid rise in obesity over the past few decades comes from a frequent mismatch between calorie intake and physical activity. Physical activity can be monitored with wearable devices such as pedometers. However, no corresponding solutions are available for automated monitoring of food intake.

The 24-hour dietary recall and food record methods are approaches for assessing food and nutrient intake that are commonly used in nutrition research and surveillance [51]. The 24-hour dietary recall method involves asking a person to recall and report all foods and beverages consumed the previous day. The conventional recall interview is carried out by a trained interviewer, while newer web-based recall methods are completed by self-administration. The food record method involves asking a person to record all foods and beverages consumed over a day, either on paper or via mobile phone apps.

The 24-hour dietary recall and food record methods have numerous strengths in comparison to other self-report methods of dietary assessment such as food frequency questionnaires, making them the current preferred tools for monitoring food and nutrient intake of populations and studying diet-disease associations [52]–[54]. However, there are important limitations. Most notably, under-estimation of energy intake is a pervasive problem [53], [55], with the magnitude of under-estimation greater among those who are overweight or obese [53], [56], female [57], and less educated [56]. Error in estimating food and nutrient intake has adverse implications for nutrition research and surveillance [58], most notably the attenuation of diet related disease risks.

Multiple factors are believed to contribute to error in reporting with the dietary recall and food record methods, including memory failure, social desirability bias, measurement reactivity, participant fatigue, and difficulty estimating food amounts [59]. The task of recording food consumption could be simplified if eating occasion data could be automatically recorded with a wearable device. There are many commercial devices on the market that monitor physical

activity, but no corresponding solutions are currently available for automated monitoring of food intake. Thus, there is clear need for a food intake monitoring (FIM) system that can automatically record food consumption for the community of patients that need diet assessment.

The current and subsequent chapters of the thesis develops a wearable FIM system initially reported in [38] that can automatically record food consumption for the community of patients that need diet assessment. The two primary objectives of the FIM system are to first detect food consumption by the subject whenever it occurs in order to record the time of eating, and second to capture a camera image of the food during each detected consumption event. The current chapter focuses on the first task of eating detection and the following chapter will focus on automated image capture.

We consider two hardware configurations of the FIM system. The first configuration will consist of a wrist band and an upper arm band, both of which are instrumented with sensors. The two bands communicate wirelessly with a computer in the home to synchronize, process, and enable backup of data. By monitoring hand position relative to the head, the system detects when the user is placing his/her hand near the mouth, a necessary condition for eating. The second configuration of the FIM device uses a single sensor worn on the wrist along with a magnet worn on the torso. By further utilizing machine learning tools, the activity is classified as eating or not eating.

The first step in detecting eating is estimating the hand position relative to the mouth. In the case of the multi sensor configuration, tracking the hand requires knowledge of the spatial orientation of each FIM sensor. As discussed in the introductory material, the 3D orientation of each sensor can be parameterized in a number of ways, including Euler angles, quaternions, and the 3x3 direction cosine matrix (DCM). Euler angles suffer from singularity issues [5], and are not suitable for estimation algorithms involving large rotations, such as arm movement. Quaternions do not suffer from these singularity issues, but the kinematic model describing their time evolution is non-linear, thus attitude filtering using a quaternion parameterization is often accomplished with the Extended Kalman Filter (EKF), or Unscented Kalman Filter (UKF) [60] which do not guarantee global stability. Although it contains redundancies, the DCM does not suffer from singularities and the kinematic model is linear, thus allowing the use of a linear Kalman Filter [61], guaranteeing convergence. Once the orientation of each sensor is known, the real-time position of the hand can be computed using a forward kinematic equation. In the case of the single sensor configuration, a simple threshold condition of the measured magnetic field is used to detect hand-to-mouth proximity. Thus, in the single configuration, detailed tracking of the

hand position is sacrificed for a simpler hardware architecture. However, both configurations answer the question of interest: “Is the hand near the mouth?”

However, hand position in and of itself does not provide full information on what activity is taking place. Thus, we need a Machine Learning (ML) binary classifier to determine if the subset of activities in which the hand is near the mouth are eating. In this thesis, a support vector machine (SVM) classifies hand activity taking place near the mouth as eating or not-eating. By limiting classification only to hand activity near the mouth, training can be restricted to a smaller subset of activities (eating, teeth brushing, chin scratching, nose picking, etc). That is, the alternative approach would be to skip the hand position estimation step entirely, but that would require training a classifier to detect relatively rare events. Eating is a relatively small portion of the day, and so a classifier that detects eating against all other daily activity would achieve very high accuracy by simply always classifying an activity as non-eating. Indeed, training a classifier in such a situation requires specialized techniques to prevent the classifier from being overly biased to the majority class [62]. By first deciding if the hand is near the mouth using some form of tracking or proximity detection, we provide the eating classifier with a dataset that is naturally more balanced towards eating, which takes some burden off of the classifier.

The remainder of this chapter is organized as follows. In Section II, the FIM system prototype hardware is detailed. In Section III, the use of the DCM for estimating hand position is described. In Section IV, experimental validation showing hand position estimation and hand-mouth proximity detection is presented. In Section V, the single sensor configuration is described. In Section VI, a SVM is utilized to classify activity as eating or not-eating.

Methods and results

Wearable sensor system prototype hardware

The FIM sensor system consists of two wireless, battery-powered sensor assemblies, one each on the wrist and the upper arm. Each assembly contains a nine-axis Inertial Measurement Unit (IMU, InvenSense MPU-9250), consisting of a three-axis accelerometer, magnetometer, and gyroscope. The prototype system is shown in Fig. 30. The data from the IMU is sampled over an I^2C bus by a Bluetooth LE System on Chip (Nordic Semiconductor nRF51), which uses a 32-Bit ARM Cortex M0 CPU. Each assembly is powered by a 500 mAH lithium polymer battery. The assembly is packaged in a 3D printed housing and attached to a wearable sleeve.

In the prototype used in this chapter, raw motion data consisting of two sets of IMU data is sent to the host computer at a 50Hz rate. The battery allows the device to send a constant data

stream to the host for 36 hours before requiring charging. The host records raw data, estimates calibration parameters, estimates the proximity of the hand to the mouth, and detects whether eating events are occurring.

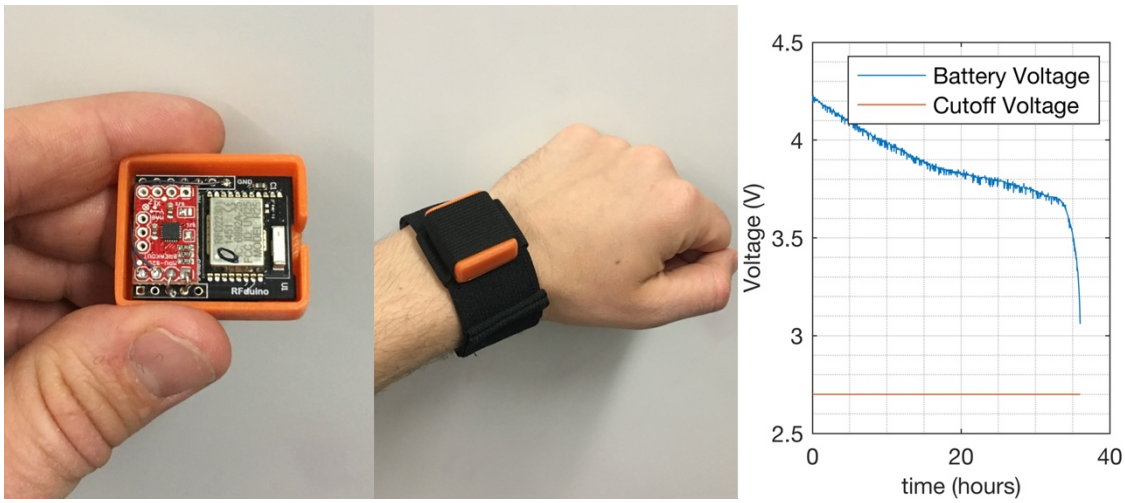

Fig. 30. FIM hardware setup
Images show the prototype wrist sensor in the Food Monitoring System (FIM). Photo on the left shows the electronics housed in a 3D printed enclosure. Plot on the right shows battery voltage during constant streaming of data, demonstrating the system can easily operate for a full day of activity before charging.

Algorithm and system architecture overview

The FIM system is broken up into a series of decisions and actions that culminate with a capture of the image of food being consumed. The focus of the current chapter is on the first two steps: detection of hand-to-mouth proximity, and classification of such activities as eating or not. We will present two methods for detecting hand proximity: the first method is a full 3D hand tracking method which requires multiple IMUs, and the second is a simpler method that detects proximity only using a single IMU and permanent magnet.

DCM Estimation

Orientation in 3D can be parameterized in a number of different ways. The three most common parameterizations are Euler angles, quaternions, and the full 3x3 DCM [5]. Euler angles consist of a series of three successive rotations about a specified order of axes. For the particular order of yaw(ψ)—pitch(θ)—roll(ϕ), a rotation that transforms a vector with representation $\boldsymbol{v}_n$ in the inertial frame to its representation $\boldsymbol{v}_b$ in the body frame can be written as follows [5]:

$$\boldsymbol{v}_b = R(\phi)R(\theta)R(\psi)\boldsymbol{v}_n \tag{58}$$

The matrices $R(\phi)$, $R(\theta)$, and $R(\psi)$ are given by:

$$R(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & c(\phi) & s(\phi) \\ 0 & -s(\phi) & c(\phi) \end{bmatrix}, \qquad R(\theta) = \begin{bmatrix} c(\theta) & 0 & -s(\theta) \\ 0 & 1 & 0 \\ s(\theta) & 0 & c(\theta) \end{bmatrix},$$
$$R(\psi) = \begin{bmatrix} c(\psi) & s(\psi) & 0 \\ -s(\psi) & c(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{59}$$

where $c(\theta)$ and $s(\theta)$ refer to cosine and sine, respectively. The full DCM is given by:

$$R_n^b = R(\phi)R(\theta)R(\psi) \tag{60}$$

When the multiplication of the individual Euler rotations is carried out the DCM can be written as:

$$R_n^b = [\boldsymbol{v}_1 \quad \boldsymbol{v}_2 \quad \boldsymbol{v}_3],$$
$$v_1 = \begin{bmatrix} c(\psi)c(\theta) \\ c(\psi)s(\phi)s(\theta) - c(\phi)s(\psi) \\ s(\phi)s(\psi) + c(\phi)c(\psi)s(\theta) \end{bmatrix},$$
$$v_2 = \begin{bmatrix} c(\theta)s(\psi) \\ c(\phi)c(\psi) + s(\phi)s(\psi)s(\theta) \\ c(\phi)s(\psi)s(\theta) - c(\psi)s(\phi) \end{bmatrix}, \qquad v_3 = \begin{bmatrix} -s(\theta) \\ c(\theta)s(\phi) \\ c(\phi)c(\theta) \end{bmatrix} \tag{61}$$

The singularity issue of the Euler angle parameterization is apparent when $\theta = \pm 90°$ [5]. In the $\theta = 90°$ case, the DCM becomes:

$$R_n^b = \begin{bmatrix} 0 & 0 & -1 \\ s(\psi - \phi) & c(\psi - \phi) & 0 \\ c(\psi - \phi) & -s(\psi - \phi) & 0 \end{bmatrix} \tag{62}$$

In other words, the ψ and ϕ angles are underdetermined, which will cause issues for estimation algorithms. A better approach is to directly estimate the DCM. The remainder of this section presents the DCM kinematic model and estimation algorithm.

DCM Process Model

The true kinematic differential equation for the DCM is [63]:

$$\dot{R}_n^b = -S(\boldsymbol{\omega})R_n^b \tag{63}$$

where $R_n^b \in \{R \in \mathbb{R}^{3\times 3} | R^{-1} = R^T, \det R = 1\}$ is a member of the Special Orthogonal rotation group (SO(3) for short) and $S(\boldsymbol{\omega})$ is a skew symmetric matrix of angular velocity in body coordinates, $\boldsymbol{\omega} \in \mathbb{R}^3$, defined as follows:

$$S(\boldsymbol{\omega}) = \begin{bmatrix} 0 & -\omega_z & -1 \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \tag{64}$$

It follows that the vectors that make up the DCM follow the same kinematic differential equation:

$$\dot{\boldsymbol{v}}_i = -S(\boldsymbol{\omega})\boldsymbol{v}_i, \; i = 1,2,3 \tag{65}$$

In reality, the true angular velocity is not known but is measured by the gyroscope, which can be modeled after removal of any bias as:

$$\boldsymbol{\omega} = \boldsymbol{\omega}_n + \boldsymbol{\varepsilon} \tag{66}$$

where $\boldsymbol{\varepsilon} \sim N\left(0, \sigma_g^2 I_3\right)$ is a vector of zero mean Gaussian noise and ω_n is the true angular velocity. Assuming constant angular velocity between samples results in the discretization:

$$\boldsymbol{v}_i[k+1] = \exp(-S(\boldsymbol{\omega}_k - \boldsymbol{\varepsilon}_k)\Delta t)\, \boldsymbol{v}_i[k] \tag{67}$$

The matrix exponential is written in power series form as [64]:

$$\exp(-S(\boldsymbol{\omega}_k - \boldsymbol{\varepsilon}_k)\Delta t) = I_3 - S(\boldsymbol{\omega}_k)\Delta t + S(\boldsymbol{\varepsilon}_k)\Delta t + h.o.t. \tag{68}$$

Neglecting higher order terms involving powers of Δt and $S(\boldsymbol{\varepsilon}_k)$ greater than two gives the approximation:

$$\exp(-S(\boldsymbol{\omega}_k - \boldsymbol{\varepsilon}_k)\Delta t) \cong \exp(-S(\boldsymbol{\omega}_k)\Delta t) + S(\boldsymbol{\varepsilon}_k)\Delta t \tag{69}$$

Finally the discretized kinematic equation can be written as:

$$\boldsymbol{v}_i[k+1] = \exp(-S(\boldsymbol{\omega}_k)\Delta t)\, \boldsymbol{v}_i[k] + S(-\boldsymbol{v}_i[k])\Delta t \boldsymbol{\varepsilon}_k \tag{70}$$

To get the DCM process dynamics into a Kalman Filter framework, we vectorize the DCM by stacking $\boldsymbol{v}_1$, $\boldsymbol{v}_2$, and $\boldsymbol{v}_3$ into the state vector $\boldsymbol{x} = [\boldsymbol{v}_1^T \quad \boldsymbol{v}_2^T \quad \boldsymbol{v}_3^T]^T$, such that the process dynamic and noise model can be written as [61]:

$$\boldsymbol{x}_{k+1} = F_k \boldsymbol{x}_k + \boldsymbol{w}_k$$

$$F_k = \begin{bmatrix} \exp(-S(\boldsymbol{\omega}_k)\Delta t) & 0_3 & 0_3 \\ 0_3 & \exp(-S(\boldsymbol{\omega}_k)\Delta t) & 0_3 \\ 0_3 & 0_3 & \exp(-S(\boldsymbol{\omega}_k)\Delta t) \end{bmatrix}$$

$$\boldsymbol{w}_k \sim N(0, Q_k), \qquad Q_k = \begin{bmatrix} Q_{1,k} & 0_3 & 0_3 \\ 0_3 & Q_{2,k} & 0_3 \\ 0_3 & 0_3 & Q_{3,k} \end{bmatrix}$$

$$Q_{i,k} = (\Delta t)^2 S(-\boldsymbol{v}_i[k]) \sigma_g^2 S(-\boldsymbol{v}_i[k])^T \tag{71}$$

DCM measurement model

The low frequency or static component of the DCM can be fully estimated using the accelerometer and magnetometer measurements without using the gyroscopes. Typical measurement models for IMUs rely on knowledge of the local geomagnetic field vector, which varies by location on Earth. For instance, a typical model for the magnetometer would be the known local geomagnetic field vector rotated into the body frame:

$$\boldsymbol{y}_{mag}(k) = R_n^b(k)\boldsymbol{b}_n + \boldsymbol{\varepsilon} \tag{72}$$

where $R_n^b(t)$ is the sensor-to-earth DCM and $\boldsymbol{b}_n$ is the fixed local geomagnetic field, essentially a parameter found using some a priori knowledge such as a model of the geomagnetic field. Alternatively, here the assumption is made only that the magnetic field is fixed relative to gravity,

which allows the algorithm to work without prior knowledge of the magnetic field. The approach here uses accelerometer measurements of gravity to fix two of three degrees of rotational freedom of the sensor, and the magnetic measurement is only used to determine the angular rotation about the gravity vector.

First, define the inertial frame as Z aligned with gravity, X aligned with magnetic north, and Y by the right hand rule. By this convention, there is no magnetic field in the inertial Y direction [61]. Because the Z direction is down, the DCM component v_3 can be determined exclusively by the accelerometer with the assumption that motion is slow and the only measured acceleration is gravity. Specifically, assume the accelerometer measurement is a noisy measurement of gravity (in units of g's) in the body frame:

$$\boldsymbol{y}_{acc}(k) = \boldsymbol{g}_b(k) + \boldsymbol{\varepsilon}_{acc}(k) \tag{73}$$

Substitute for $\boldsymbol{g}_b$ using the DCM:

$$\boldsymbol{y}_{acc}(k) = R_n^b(k)\boldsymbol{g}_n + \boldsymbol{\varepsilon}_{acc}(k) \tag{74}$$

Note, by the Z-aligned gravity definition, $\boldsymbol{g}_n = [0 \quad 0 \quad 1]^T$, which implies the accelerometer is a noisy measurement of the third column of the DCM:

$$\boldsymbol{y}_{acc}(k) = \boldsymbol{v}_3 + \boldsymbol{\varepsilon}_{acc}(k) \tag{75}$$

The magnetic field has only X and Z components by our chosen convention, and thus the magnetic field in the inertial frame can be written as:

$$\boldsymbol{b}_n = \begin{bmatrix} B_x \\ 0 \\ B_z \end{bmatrix} \tag{76}$$

Thus, the magnetic field in the body frame, $\boldsymbol{b}_b$, at time k is given by the transformation:

$$\boldsymbol{b}_b(k) = \begin{bmatrix} b_x(k) \\ b_y(k) \\ b_z(k) \end{bmatrix} = R_n^b(k) \begin{bmatrix} B_x \\ 0 \\ B_z \end{bmatrix} \tag{77}$$

Noting that $\left(R_n^b\right)^{-1} = \left(R_n^b\right)^T$ by definition, then it follows from above that:

$$B_Z = \boldsymbol{v}_3^T \begin{bmatrix} b_x(k) \\ b_y(k) \\ b_z(k) \end{bmatrix} \tag{78}$$

Noting that $\boldsymbol{v}_3$ was found with the accelerometer, and so B_Z can be directly solved for. Given B_Z, we can also solve for B_x using magnitude matching, since the magnetic field magnitude is independent of the frame it is expressed in:

$$\|\boldsymbol{b}\|_2^2 = B_x^2 + B_Z^2 = b_x^2(k) + b_y^2(k) + b_z^2(k) \tag{79}$$

$$B_x = \sqrt{b_x^2(k) + b_y^2(k) + b_z^2(k) - B_Z^2} \tag{80}$$

Given our chosen inertial convention that the inertial X axis points in the direction of the geomagnetic field, then B_x is always positive. Once B_x and B_z are found, then we have all the information needed to find $\boldsymbol{v}_1$ by substituting B_x and B_z into (77), and solving for $\boldsymbol{v}_1$:

$$\boldsymbol{v}_1(k) = \left(\frac{1}{B_x}\right)\boldsymbol{b}_b(k) - B_z\boldsymbol{v}_3(k) \tag{81}$$

Finally, we may now apply the orthonormal constraint on R_n^b to find $\boldsymbol{v}_2$:

$$\boldsymbol{v}_2(k) = \boldsymbol{v}_3(k) \times \boldsymbol{v}_1(k) \tag{82}$$

To summarize, we have just shown how to transform the magnetometer and accelerometer measurements into the three column vectors that make up the DCM R_n^b. To summarize:

$$\boldsymbol{x} = \begin{bmatrix}\boldsymbol{v}_1\\ \boldsymbol{v}_2\\ \boldsymbol{v}_3\end{bmatrix} = \boldsymbol{f}(\boldsymbol{y}_n) = \begin{bmatrix} \left(\boldsymbol{b}^t\boldsymbol{b} - \frac{(\boldsymbol{a}^T\boldsymbol{b})^2}{\|\boldsymbol{a}\|_2^2}\right)^{-\frac{1}{2}}\left(\boldsymbol{b} - \left(\frac{\boldsymbol{a}^T\boldsymbol{b}}{\|\boldsymbol{a}\|_2^2}\right)\boldsymbol{a}\right) \\ \left[\frac{\boldsymbol{a}}{\|\boldsymbol{a}\|_2}\right]_\times \left(\left(\boldsymbol{b}^t\boldsymbol{b} - \frac{(\boldsymbol{a}^T\boldsymbol{b})^2}{\|\boldsymbol{a}\|_2^2}\right)^{-\frac{1}{2}}\left(\boldsymbol{b} - \left(\frac{\boldsymbol{a}^T\boldsymbol{b}}{\|\boldsymbol{a}\|_2^2}\right)\boldsymbol{a}\right)\right) \\ \frac{\boldsymbol{a}}{\|\boldsymbol{a}\|_2} \end{bmatrix},$$

$$\boldsymbol{y}_n = \begin{bmatrix}\boldsymbol{a}\\ \boldsymbol{b}\end{bmatrix} = [a_x \quad a_y \quad a_z \quad b_x \quad b_y \quad b_z]^T \tag{83}$$

where $[\boldsymbol{p}]_\times\boldsymbol{q} \equiv \boldsymbol{p} \times \boldsymbol{q}$. In reality, we have noisy accelerometer and magnetometer measurements modeled as:

$$\boldsymbol{y} = \boldsymbol{y}_n + \boldsymbol{\varepsilon}, \qquad \boldsymbol{\varepsilon} \sim N\left(0, blkdiag\left(\sigma_{acc}^2 I_3, \sigma_{mag}^2 I_3\right)\right) \tag{84}$$

Substitute the above equation into the function that transforms the true acceleration and magnetometer measurements into the columns of the DCM:

$$\boldsymbol{x} = \begin{bmatrix}\boldsymbol{v}_1\\ \boldsymbol{v}_2\\ \boldsymbol{v}_3\end{bmatrix} = \boldsymbol{f}(\boldsymbol{y} + \boldsymbol{\varepsilon}) \tag{85}$$

Note, we have ignored the sign on ε without loss of generality, as it is zero mean Gaussian. Treat $\boldsymbol{\varepsilon}$ as a small perturbation around $\boldsymbol{y}$ and utilize the first order Taylor expansion:

$$\begin{bmatrix}\boldsymbol{v}_1\\ \boldsymbol{v}_2\\ \boldsymbol{v}_3\end{bmatrix} \approx \boldsymbol{f}(\boldsymbol{y}) + J_f(\boldsymbol{y})\boldsymbol{\varepsilon} \tag{86}$$

where $J_f(\boldsymbol{y})$ is the Jacobian of $\boldsymbol{f}$ evaluated at measurement $\boldsymbol{y}$. Recall that an affine transformation of a zero mean Gaussian random variable with covariance P results in a zero mean Gaussian random variable with covariance APA^T, where A is the affine transformation. Thus, given noisy

accelerometer and magnetometer measurements $\boldsymbol{y}$, we transform into measurements $\boldsymbol{z}$ of the form:

$$\boldsymbol{z} = \boldsymbol{f}(\boldsymbol{y}) \approx H\boldsymbol{x} + \boldsymbol{\eta},$$
$$H = I_9, \qquad \boldsymbol{\eta} \sim N\left(0, J_f(\boldsymbol{y})\left(blkdiag\left(\sigma_{acc}^2 I_3, \sigma_{mag}^2 I_3\right)\right)J_f^T(\boldsymbol{y})\right),$$
$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{v}_1 \\ \boldsymbol{v}_2 \\ \boldsymbol{v}_3 \end{bmatrix} \tag{87}$$

Orthonormality constraint

Before combining the DCM process and measurement models into a Kalman Filter, we address an issue with loss of orthonormality in the Kalman update step. Suppose the prior state vector, $\hat{\boldsymbol{x}}_k^-$ is made up of stacked orthonormal vectors as described above. When applying a Kalman update of the form:

$$\hat{\boldsymbol{x}}_k^+ = \hat{\boldsymbol{x}}_k^- + K(\boldsymbol{z}_k - H\hat{\boldsymbol{x}}_k^-) \tag{88}$$

there is no constraint guaranteeing orthonormality in the posterior estimate and over time we may run into issues with the loss of orthonormality. So, we wish to find the estimate $\hat{\boldsymbol{x}}_k^{++}$ comprised of orthonormal vectors "closest" to the unconstrained estimate $\hat{\boldsymbol{x}}_k^+$. Letting $\hat{R}_k^{++}$ denote the constrained posterior DCM corresponding to $\hat{\boldsymbol{x}}_k^{++}$, and $\hat{R}_k^+$ the unconstrained estimate, the closest constrained estimate to the unconstrained estimate in the sense of the Frobenius norm is given by the minimum to the loss function [5]:

$$L\left(\hat{R}_k^{++}\right) = \left\|\hat{R}_k^{++} - \hat{R}_k^+\right\|_F^2$$
$$s.t.\ \hat{R}_k^{++} \in \{R \in \mathbb{R}^{3\times 3} | R^{-1} = R^T, \det R = 1\} \tag{89}$$

The above can be shown to be equivalent to Wahba's problem (itself a special case of the more general Procustes problem), which finds the optimal rotation matrix which minimizes the error in the least squares sense between a set of vector observations in the body frame to their known representations in the reference frame rotated by the aforementioned rotation matrix [5]. Several analytical solutions have been found to the problem, and in the case of DCM estimation, a solution using the singular value decomposition (SVD) is most appropriate. Let the SVD of $\hat{R}_k^+$ be given by:

$$\hat{R}_k^+ = V\Sigma U^T \tag{90}$$

then the closest DCM in the sense of the Frobenius norm is given by:

$$\hat{R}_k^{++} = V diag(I_2, \det V \det U) U^T \tag{91}$$

Intuitively, we can think of the SVD as a rotation followed by a scaling of the matrix of singular values, Σ, followed by a second rotation. Thus, the closest rotation matrix to the unconstrained

estimate simply removes the scaling, as proper rotations preserve vector magnitudes. The $diag(I_2, \det V \det U)$ term guarantees a +1 determinant in the constrained rotation matrix.

DCM Kalman Filter

We now summarize the previous sections and present the DCM Kalman filter (KF). The KF noisy process model is given by:

$$\boldsymbol{x}[k+1] = F[k]\boldsymbol{x}[k] + \boldsymbol{w}[k] \tag{92}$$

where the state vector $\boldsymbol{x} \in \mathbb{R}^9$ is the stacked three column vectors of the DCM R_n^b. The state transition matrix $F[k]$ is determined from the gyroscope measurements along with process noise covariance Q per (71).

The KF noisy measurement model is given by:

$$\boldsymbol{z}[k] = H\boldsymbol{x}[k] + \boldsymbol{\eta}[k] \tag{93}$$

where $H = I_9$ and $\boldsymbol{z}[k]$ is a vector of transformed accelerometer and magnetometer measurements per (83) with measurement noise covariance per (87).

The standard KF equations apply, with the addition of the orthonormal constraint step after each measurement update. Let $\hat{R}_k^+$ be the posterior estimate immediate in matrix form after the measurement update:

$$\hat{R}_k^+ \equiv \begin{bmatrix} \hat{\boldsymbol{x}}_k^+(1) & \hat{\boldsymbol{x}}_k^+(4) & \hat{\boldsymbol{x}}_k^+(7) \\ \hat{\boldsymbol{x}}_k^+(2) & \hat{\boldsymbol{x}}_k^+(5) & \hat{\boldsymbol{x}}_k^+(8) \\ \hat{\boldsymbol{x}}_k^+(3) & \hat{\boldsymbol{x}}_k^+(6) & \hat{\boldsymbol{x}}_k^+(9) \end{bmatrix} \tag{94}$$

where $\boldsymbol{x}(i)$ is the ith element of $\boldsymbol{x}$. Find the SVD, $\hat{R}_k^+ = V\Sigma U^T$ and then constrain the posterior estimate to be orthonormal:

$$\hat{R}_k^+ \equiv \begin{bmatrix} \hat{\boldsymbol{x}}_k^{++}(1) & \hat{\boldsymbol{x}}_k^{++}(4) & \hat{\boldsymbol{x}}_k^{++}(7) \\ \hat{\boldsymbol{x}}_k^{++}(2) & \hat{\boldsymbol{x}}_k^{++}(5) & \hat{\boldsymbol{x}}_k^{++}(8) \\ \hat{\boldsymbol{x}}_k^{++}(3) & \hat{\boldsymbol{x}}_k^{++}(6) & \hat{\boldsymbol{x}}_k^{++}(9) \end{bmatrix} = V diag(I_2, \det V \det U) U^T \tag{95}$$

Hand position estimation using multiple IMUs

By using the constrained KF on two IMUs worn on the upper and lower arms, the position of the wrist joint, P_w, relative to an origin at the shoulder can be estimated using the upper and lower arm lengths, r_1 and r_1, respectively:

$$P_w = R_{b1}^n \begin{bmatrix} r_1 \\ 0 \\ 0 \end{bmatrix} + R_{b2}^n \begin{bmatrix} r_2 \\ 0 \\ 0 \end{bmatrix} \tag{96}$$

Essentially, we treat the upper and lower arms as rigid links and apply a forward kinematic equation. Note, that the coordinates above are oriented relative to the inertial frame with its origin

at the shoulder, so it only partially captures our desired task of determining hand-to-mouth proximity.

To fully constrain the position of the hand relative to the mouth, we need to consider the joint angles of all body segments in between the hand and mouth. That is, we need to know the head, torso, upper arm, lower arm, and hand orientations along with each limb's length. Tracking the hand position to such a level of detail would require five IMUs, one on each of the listed body segments. Such an arrangement is impractical and undesirable from the application perspective. With a third IMU on the torso, we can track the hand relative to the torso in body-fixed coordinate axes (rather than inertial-fixed axes). As an example, we can clearly see experimentally [38] that three IMUs is adequate to visually discern arm motions such as eating, waving, and horizontally sweeping the hands, as shown in Fig. 31.

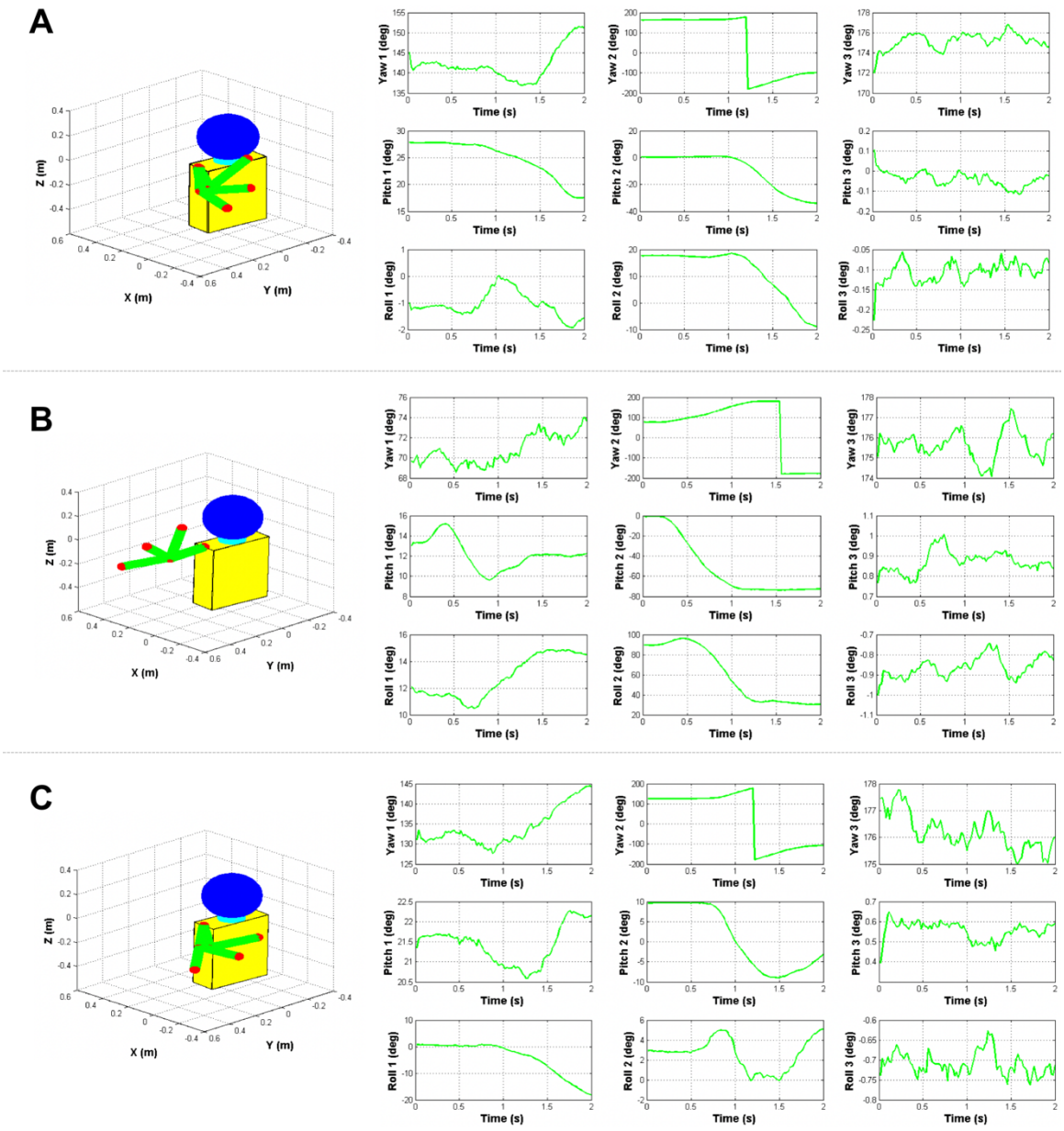

Fig. 31. Experimental arm tracking
Results of arm tracking with three IMUs: one on the torso, one on the upper arm, and one on the lower arm.

Even limiting the system to three IMUs is likely impractical from the FIM application perspective. With two IMUs, we can determine that the base of the hand is at a certain height and radius from the shoulder, which is in and of itself a necessary condition for eating. With certain additional constraints added regarding the range of motion of each joint, the hand position could be constrained further. Thus, even two IMUs provides helpful information regarding necessary conditions for eating and is arguably a practical hardware arrangement. But even two IMUs is

less than desirable as each IMU requires an independent microcontroller, battery supply, and wireless synchronization. We next present a simpler approach to the problem using a single wrist-worn IMU.

Hand-to-mouth proximity estimation using a single sensor

Recall, the intention in the first step of eating detection is to determine hand-to-mouth proximity as a necessary condition for eating. In this section, we consider a much simpler manner of determining hand proximity by taking a different approach entirely. As we have seen above, estimating the 3D coordinates of the hand is a challenging endeavor with several pitfalls. Even with good orientation estimation algorithms, we're still relying on the geomagnetic field to fully constrain orientation in indoor settings, a task well known to be rife with problems as discussed in the introductory material of this thesis.

In this section, we dispense with tracking the hand in 3D coordinate space and instead directly answer the hand proximity question using the magnetometer alone. Consider the magnetic field of a dipole in cylindrical coordinates [65]:

$$B_r = \frac{\mu_0 dm_0}{2\pi r^3} cos(\alpha),$$
$$B_\alpha = \frac{\mu_0 dm_0}{4\pi r^3} sin(\alpha) \tag{97}$$

where μ_0 and dm_0 are constant parameters, r is the radius from the dipole center, and α is the angle from the dipole axis. If we consider the magnitude of the field at any point as a function of r and α, we have:

$$b(r, \alpha) = \frac{\mu_0 dm_0}{2\pi r^3} \sqrt{\frac{1}{4}\sin^2\alpha + \cos^2\alpha} \tag{98}$$

The magnitude of the magnetic field has an inverse cube relationship with the radius, and at any given radius, the field varies by a factor of 2 as a function of alpha (the term under the square root varies from 1/2 to 1). This nonlinearity suggests we might use a permanent magnet on the body as a means of hand-to-mouth proximity detection. If a magnet can be secured to the body such that the IMU is roughly aligned with the axis of the magnet and in close proximity when the hand is close to the mouth, then we would expect the IMU to measure a strong magnetic field magnitude when the hand is close to the mouth that would drop off steeply when the hand is elsewhere. The magnetic field measured by a three-axis magnetometer $\boldsymbol{y}_{mag} \in \mathbb{R}^3$ with a magnet on the torse can be modeled as the magnitude of the vector sum of several terms:

$$\boldsymbol{y}_{mag} = \boldsymbol{b}_{magnet}(\boldsymbol{p}, R^b_{torso}) + R^b_n \boldsymbol{b}_{earth} + \boldsymbol{\varepsilon}_{bias} + \boldsymbol{\varepsilon}_{noise} \tag{99}$$

where b_{earth} is the geomagnetic field in the inertial frame, b_{magnet} is the measured field from the magnet, which depends of the orientation R^b_{torso} and position $\boldsymbol{p}$ of the limb relative to the torso, and $\boldsymbol{\varepsilon}_{bias}$ and $\boldsymbol{\varepsilon}_{noise}$ are bias and noise terms.

Ostensibly, we could use the magnet to estimate the position of the sensor relative to the torso, allowing us to answer whether the hand is near the mouth. But directly estimating position in this manner is an exercise fraught with pitfalls. The bias term can be removed using calibration based on the assumption that the geomagnetic field magnitude is constant regardless of the orientation it is measured in. However, in practice the class of IMUs considered in this thesis may have issues with bias stability, requiring both careful and repeated calibration. Even in cases with high quality sensors, good calibration, and constraints placed on sensor rotation, estimating position using a permanent magnet as a reference has historically achieved mixed results requiring great efforts such as particle filters running alongside experimentally-trained neural network measurement models [66]. Such an approach is practically infeasible for the FIM device.

Here, we simply consider if the magnetic field from the permanent magnet dwarfs all of the other terms when the hand is near the mouth. If so, a simple threshold can be utilized to determine hand-mouth proximity without any knowledge of the sensor orientation or required calibration. To determine if such an approach is feasible, we ran a series of experiments in which the IMU was worn on the wrist and a magnet was worn on the torso near the sternum. While measuring the magnetic field, the hand was placed in a variety of positions and the body was positioned in a variety of orientations relative to gravity. An image of the experimental setup is shown in Fig. 32. Fig. 33 reports the minimum and maximum magnetic field recorded by the sensor for each hand position.

Fig. 32. Permanent magnet for hand-to-mouth proximity detection
Experimental setup of a permanent magnet used to determine hand-to-mouth proximity. When the hand is near the mouth, the IMU is directly anterior to the magnet and roughly in line with the axis of the magnet.

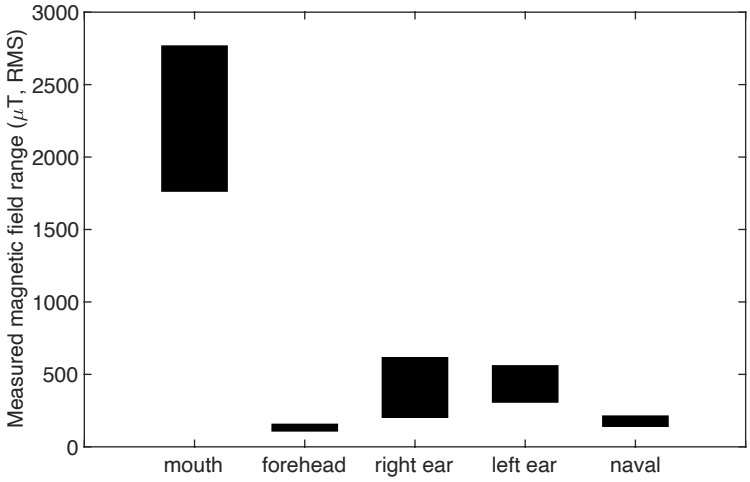

Fig. 33. Magnetic field magnitude range measured by wrist-worn FIM
Data taken at various hand positions and body orientations with a permanent magnet worn on the torso. Note: at least one axis was saturated in all measurements taken at the mouth.

As Fig. 33 demonstrates, when the hand is near the mouth, the minimum measured magnetic field was three times the maximum field measured with the hand at other positions on the body. The actual difference may have been greater, as the sensor was saturated in at least one axis for all measurements in which the hand was near the mouth. Interestingly, simply considering the Z-axis measurement was enough to discern if the hand was near the mouth. It also provides additional information beyond the magnitude, as the sign of the Z-axis measurement indicates if the palm is facing towards the mouth (a negative measurement) or away from the mouth (a positive measurement). For all measurements, the Z-axis measurement was saturated when the hand was at the mouth, suggesting a smaller magnet would be adequate for detecting hand-to-mouth proximity.

Additional information can be used from the gyroscope to increase confidence that a detected spike in the measured magnetic field is indeed the motion of the hand moving to the mouth. Fig. 34 shows gyroscope measurements taken as the hand picks up food with a fork and with the fingers and places it in the mouth. There is a noticeable and repeatable rotation involving supination of the wrist. Thus, the FIM can look for both a peak in the magnetometer signal along with wrist supination to detect hand-to-mouth proximity.

The experimental results confirm that the simpler single wrist FIM sensor can reliably detect hand-to-mouth proximity using simple threshold methods as the initial step towards detecting eating. We next consider if the IMU signals from a single wrist worn sensor can provide enough information to discern eating from non-eating activities once hand-to-mouth proximity is detected.

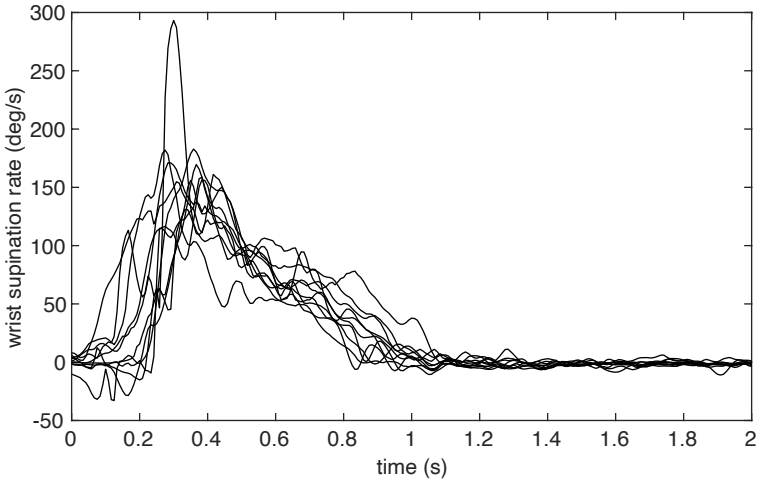

Fig. 34. Wrist supination rate as the hand goes from the plate to the mouth
Data taken from 10 experiments with equal number of bites using the fingers and a fork.

Eating detection using support vector machine

In this section, we develop a method to detect eating once hand-to-mouth proximity has been established. The hand being near the mouth is a necessary condition for eating but does not guarantee that the activity is indeed eating. To reduce false-positive eating detection, machine learning can be used to improve FIM performance by classifying activities in which the hand is near the mouth. The advantage of this approach is that the machine learning algorithm only needs to be trained on activity that meets the proximity requirement described previously. Without first estimating hand proximity, the algorithm needs to be trained against all possible hand activity. Thus, the classification of non-eating activity is a small subset of the total daily arm activities, such as shaving, brushing teeth, drinking, scratching chin, nose picking, etc.

To test the effectiveness of using machine learning tools to classify hand activity near the mouth, experimental data was taken using a wrist-worn IMU. A total of 25 minutes of data was collected while eating, shaving, and brushing teeth. The labelled data was divided into 30 second long segments, and several features were extracted, including the mean and RMS value of acceleration, the values and frequencies of the first five peaks of the acceleration power spectral density (PSD), and the power in the 0.5-1.5 Hz, 1.5-5.0 Hz, 5.0-10.0 Hz, 10.0-15.0 Hz, and 15.0-20.0 Hz bands for each sensor axis. Fig. 35 shows how the acceleration PSD in the direction of the lower arm varies for different activities. Teeth brushing has well defined peaks in the 5-15Hz range. Both teeth brushing and shaving are higher power than eating at lower frequencies. Fig. 36 shows how the extracted features separate the three activities in this experiment. In particular, the RMS acceleration provides good feature separation.

The extracted features were used to train a Support Vector Machine (SVM) to classify the 30 second segments as eating or not-eating. The SVM fits the maximum margin hyperplane in the feature space by maximizing the distance between the hyperplane and the nearest data points [67]. The SVM was trained using all data and validated by 5-fold cross validation [68]. In k-fold cross validation, the samples are randomly divided into k folds. One fold is held out, and the remaining k-1 folds are used to train an SVM. This process is then repeated a total of k times, such that each group gets used once as the test group. The overall accuracy of the SVM trained using all of the data is estimated by taking the mean accuracy of the k SVMs. For the 25-minute motion dataset, the SVM achieved 98% classification accuracy.

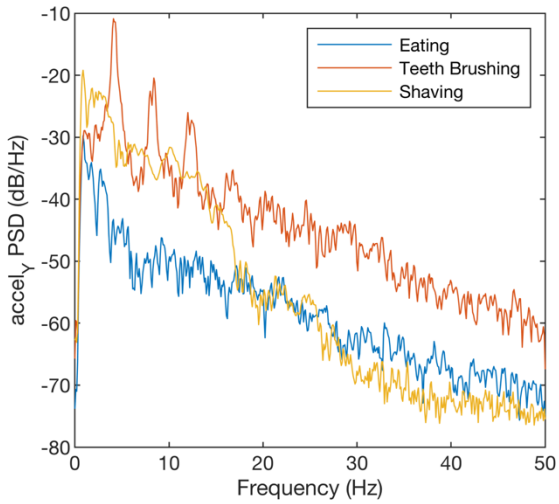

Fig. 35. Lower arm PSD for hand-near-mouth activities
Plot of acceleration PSD in direction of lower arm for three activities in which the hand is near the mouth: eating, teeth brushing, and shaving. Brushing has well defined peaks in the 5-15Hz range. Both brushing and shaving have high power at lower frequencies.

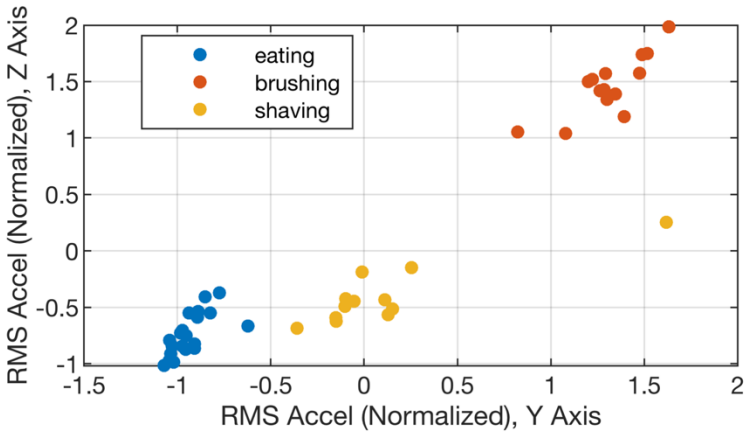

Fig. 36. Features for eating detection SVM
Example of two features used for classification in the SVM, the RMS acceleration in y and z directions. In these experiments, the y direction is along the length of the arm and z is normal to the palm when the wrist is straight. Units are normalized by subtracting the mean and dividing by the standard deviation.

Chapter 6:
Image capture and classification for a food intake monitoring device using IMUs

Introduction

In the previous chapter and in related publication [38], we have shown that motions in which the hand is near the mouth can be accurately classified as eating or not eating using features extracted from the motion data obtained from an IMU worn on the wrist. This result is the foundation of the Food Intake Monitor (FIM) device, a novel wearable sensor that automatically detects when the user is eating. As mentioned, the hand being in close proximity to the mouth is a necessary condition for eating, and the hand position relative to the torso may be fully or partially constrained using IMUs attached to all or some of the torso, upper arm, and lower arm body segments using a simple forward kinematic equation with known lengths of the body segments. We also showed that a much simpler wrist-only device may be used to determine hand-mouth proximity in conjunction with a magnet worn on the body near the collar bone. This simpler setup is preferable from the perspectives of the user (fewer devices to wear), cost (fewer components), power consumption (fewer microcontrollers), and software (no wireless synchronization). Indeed, a wrist only device is a preferable approach to the broader space of possible FIM device hardware architectures such as smart glasses (e.g. Google glass), as wrist worn smart devices have been widely adopted by the general population, as discussed in the Introduction chapter.

In this chapter of the thesis, the FIM device will be further developed to include a camera on the wrist along with associated on-board algorithms to automatically capture an image of the food being eaten and classify the image as containing food or not. Introducing a camera to the FIM device also introduces issues concerning power consumption and privacy. One approach to capturing the food is to have the camera always on and to stream images wirelessly for off-device storage. However, this is poor design from a power consumption perspective, as the camera and wireless streaming have significantly higher power requirements than the other main components in the FIM device, notably the IMU. More importantly, an always-on camera that stores images on a cloud server is poor system architecture from a user privacy perspective. Users are likely to

object to a system constantly recording images of their daily lives, including potentially capturing views of themselves and those they interact with that would be considered risky to have stored on a remote server.

Thus, a better approach is to minimize use of the camera as much as possible and to only offload images from the device when we have reasonable confidence the image contains food. In the FIM device, the camera is only enabled once eating is detected by the previously mentioned hand-proximity and eating-detection algorithms, and timing of triggering the image capture is determined by the IMU. IMU-driven camera triggering reduces both the power consumption and privacy concerns of an always-on approach. For instance, the IMU may completely block the camera from taking images when the wrist is vertical, an orientation unlikely to contain food but likely to contain views of the user or their surroundings.

Further, performing on-device screening of the captured image to decide if it contains food prior to transferring the image off of the device provides an additional layer of privacy protection for the user and reduces wasted resources needed to transfer the image. Image classification and other computer vision tasks such as image segmentation has traditionally been limited to the purview of cloud-based systems running deep neural networks such as convolutional neural networks (CNNs) [69]. However, there has been much recent interest in so-called TinyML, in which machine learning (ML) algorithms are deployed on resource constrained devices, commonly referred to as edge inference [70]. A common application of edge inference is keyword spotting ("OK, Google" or "Hey, Siri") in which the device itself listens for special keywords indicating the user is ready to interact with the device [71]. In the TinyML paradigm, a neural network or other ML algorithm is typically trained using a system without resource constraints and then adapted through post-processing for microcontroller edge inference. As an example, a neural network can be adapted using quantization in which parameters stored using float values during training are converted into 8-bit integers with minimal loss in accuracy.

In the case of TensorFlow Lite, a small subset of the full TensorFlow operation library is available for MCU deployment, but enough operations are supported that complex networks such as deep CNNs can be built. Unlike the prototypical neural network architecture which takes a 1-dimensional input, CNNs use the inherent 2-dimensional (in the case of greyscale) structure of images. Each layer of a CNN is essentially a trainable image filter, and the network learns filters that extract features (e.g. object edges) during training. CNNs are typically trained using 32-bit floating point parameters, but recently published applications employing quantized CNNs for edge inference may be found in [72], [73].

In the case of the FIM sensor, the device follows a series of decisions before capturing images, as developed in the previous chapter. First, it finds when the hand is in close proximity to the mouth. Then, it classifies the activity as eating or not eating. If eating is detected, then the IMU is used to determine the optimal time to attempt to capture an image of the food. Finally, if an image is captured, the device checks if the image contains food using an on-device CNN. Specifically, the main contributions in this chapter are:

- The development of a novel privacy- and power- sensitive algorithm to automatically capture food being consumed using a wrist worn camera by partially constraining the real-time 3D orientation of the wrist using a Kalman Filter.
- The development of a integer-quantized convolutional neural network adapted to microcontroller deployment that classifies images as containing food (or not) without requiring any wireless transfer of images or cloud-based computing.
- Experimental verification showing the IMU-guided image capture algorithm can reliably capture images of food in front of the user and classified as containing food or not.

The remainder of this chapter is organized as follows. In the Methods section, we present the hardware used for image capture, along with development of an algorithm to optimally time the capturing of the image. We also present the CNN for image classification and the methodology contributing to its use. In the Results section, experimental results are presented demonstrating selection of the orientation for image capture and subsequent tuning of the target selection. We additionally discuss the training and accuracy of the CNN for image classification.

Methods

Hardware overview

The hardware used for this set of experiments is an Arduino Nano BLE Sense, a small 45x18 mm device consisting of a Nordic Semiconductor nRF52840 Bluetooth ARM Cortex-M4 System on Chip along with a suite of on-board sensors. Notably, the board contains a 9-axis LSM9DS1 IMU from ST Microelectronics, consisting of a tri-axis accelerometer, a tri-axis gyroscope, and a tri-axis magnetometer. The board does not contain a camera, so an external breakout board containing an OmniVision OV7675 camera was utilized. The OV7675 sensor can produce color images up to a VGA resolution of 640x480 pixels (though the Arduino does not support the highest resolution offered by the camera). An image of the experimental setup is shown in Fig.

37. The overall device size is 80x50 mm and can be attached to the wrist using an elastic band and hook-and-loop closure.

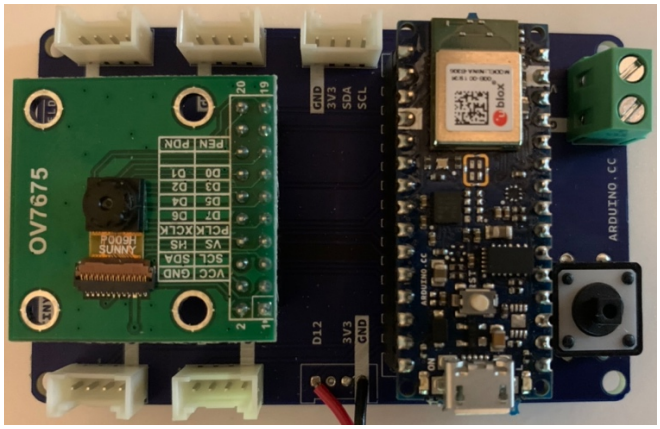

Fig. 37. FIM camera hardware
Image of the experimental hardware used for the feasibility study. Left breakout board is the OV7675 camera module. Right board is the Arduino Nano BLE Sense containing the IMU and microcontroller. Overall dimensions of the device as shown is 80x50 mm and can be worn on the wrist using an elastic strap with hook and loop-closure.

Camera triggering algorithm

Algorithm overview

The objective of the camera operating algorithm is to utilize the IMU to determine some optimal time to enable the camera and take an image while the user eats. Specifically, we will define the optimal time to take a picture as the time when the camera orientation is closest (in an angular sense) to some predetermined orientation deemed most likely to contain food. The FIM sensor has three degrees of rotational freedom, sometimes referred to as pitch, roll, and yaw Euler angles. As discussed throughout this thesis, determining the yaw angle relative to earth is challenging with mobile-phone grade IMUs in interior settings, so we will only concern ourselves with the sensor pitch and roll angles. It should be emphasized at this point that we do not yet know if constraining two of the three degrees of rotation provides enough information to find an optimal image capture point, but we aim to show this as an experimental result later in the chapter. For now, we will presume partially constraining the full orientation of the sensor provides enough information.

The remainder of this subsection will first discuss (a) how we estimate the partial orientation of the sensor using a real-time on-board Kalman Filter, (b) how we can estimate the optimal capture point of the image, and (c) how to determine the best orientation to target image capture.

Sensor partial orientation estimation

The pitch and roll angles constrain the orientation of the FIM sensor relative to gravity. While intuitive, there are issues using these angles directly. For example, if the pitch of the sensor is aligned with gravity, then the roll angle is ill-defined. Arm motions can easily pass through orientations that are problematic for Euler angles. Fortunately, the same pitch and roll information is contained within the gravity vector itself (in the sensor frame of reference), and this vector is always well-defined and requires no trigonometric functions to compute. Thus, it is the better choice for working with at the microcontroller level. The gravity vector in the sensor frame of reference is given by:

$$\boldsymbol{g}_S(t) = R_{SE}(t)\boldsymbol{g}_E \tag{100}$$

where $\boldsymbol{g}_E \in \mathbb{R}^3$ is the fixed gravity vector in the Earth frame of reference and $R_{SE}(t)$ is the time varying 3x3 Direction Cosine Matrix (DCM) defining the orientation of the FIM relative to earth. The kinematic differential equation for the gravity vector is governed by the angular velocity of the sensor, $\boldsymbol{\omega}_S(t) \in \mathbb{R}^3$:

$$\dot{\boldsymbol{g}}_S(t) = -[\boldsymbol{\omega}_S(t)]_\times \boldsymbol{g}_S(t) \tag{101}$$

where the cross product matrix $[\boldsymbol{a}]_\times$ for $\boldsymbol{a} \in \mathbb{R}^3$ is given by:

$$[\boldsymbol{a}]_\times \equiv \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \tag{102}$$

Triggering the camera off of the FIM orientation relative to gravity implies that our device must estimate the gravity vector in real-time. The accelerometer measurement can be modeled as a noisy measurement of gravity with the assumption of small linear acceleration:

$$\begin{gathered} \boldsymbol{y}_{acc}(k) = -\boldsymbol{g}_S(k) + \boldsymbol{v}(k), \\ \boldsymbol{v}(k) \sim \mathcal{N}(0, \sigma_{acc}^2 I_3) \end{gathered} \tag{103}$$

Using the acceleration measurement directly as an estimate of gravity is a poor choice, because in reality, the measurement contains linear accelerations that may sometimes be significant. Low pass filtering can be used to remove these linear accelerations, but this approach causes a phase lag which would cause problems for real-time triggering of the camera. The IMU gyroscope could conceivably be integrated to obtain the gravity vector, assuming we know the initial orientation, but this is only practical for very short durations, as bias in the gyroscope will

cause integration drift. The class of low-cost IMUs used in this work are particularly notorious for bias instability. Fortunately, the gyroscope and accelerometer may be fused in a simple linear time varying Kalman Filter, which essentially will act as a complementary filter blending the high-frequency estimate provided by the gyroscope with the low-frequency drift-free estimate provided by the accelerometer. The process equation for the Kalman Filter is given by (101), and the measurement equation is given by (103).

An issue with estimating the gravity vector in this manner is that the additive update is guaranteed to destroy the norm constraint on $\boldsymbol{g}_S$. That is, the individual elements of $\boldsymbol{g}_S$ are time varying, but the vector itself always has a 1 g norm. A practical way of handling this issue is to simply renormalize the vector after every measurement update, which will prevent the state estimate norm from drifting. While renormalization is not necessarily optimal in any sense, our previous work has shown under simulation that renormalization provides adequate norm-constrained estimates of gravity for the computational constraints of the FIM application [37].

Camera triggering algorithm

Let's suppose we have some target value of g_S which typically will put the camera in view of a plate of food after the FIM sensor has detected eating. In other words, we wish to snap a photo when $g_S \approx g_{Des}$, where g_{Des} is our desired target attitude relative to gravity. Suppose we will snap the photo when the FIM estimate of g_S is within some predetermined angle, θ_{Des}, of $\boldsymbol{g}_{Des}$. Effectively this is controlling the pitch and roll angles in which to take the photo. At any given time, t, the angle between $\boldsymbol{g}_S$ and $\boldsymbol{g}_{Des}$ is:

$$\theta(t) = acos\left(\boldsymbol{g}_{Des}^T \boldsymbol{g}_S(t)\right) \tag{104}$$

This follows from assuming the gravity vectors have unit norm and:

$$\boldsymbol{a}^T\boldsymbol{b} = \|\boldsymbol{a}\|\|\boldsymbol{b}\| \cos\theta \tag{105}$$

So, one simple way of triggering a photo is a simple if-statement:

- if $\text{acos}\left(\boldsymbol{g}_{Des}^T \boldsymbol{g}_S(t)\right) < \theta_{Des}$, then take the photo

However, this will tend to always take the photo on the boundary $\text{acos}\left(\boldsymbol{g}_{Des}^T \boldsymbol{g}_S(t)\right) = \theta_{Des}$, which is problematic. For instance, with a generous θ_{Des}, the object will never be centered in the field of view. On the other hand, if θ_{Des} is too small, we risk missing the threshold even if we could have had the object in view at some point in the trajectory. So, we might instead wait a handful of milliseconds after hitting the threshold to take the image. But, how many milliseconds

should we wait? And, shouldn't the time we wait to take the image be dependent on how "fast" or "slow" the motion is? What we really desire is something slightly more sophisticated in which we only take a photo inside some threshold, but if we predict the camera will continue to approach the object once inside the threshold, we wait to take the photo. In pseudo-code, this can be stated as:

- If $\operatorname{acos}\left(\boldsymbol{g}_{Des}^{T}\boldsymbol{g}_{S}(t)\right) < \theta_{Des}$ and θ_{Des} is decreasing, then snap a photo when we estimate $\boldsymbol{g}_{S}$ will be closest to $\boldsymbol{g}_{Des}$ given the current trajectory.

We have no way of knowing with certainty the future trajectory, but we can predict where the camera will move by integrating the IMU's gyroscope. Integrating the gyroscope allows us to estimate if our image field of view will improve and how much time it will take to get to the optimal orientation relative to the target.

First, note that cosine is monotonically decreasing in the range $0 < \theta < \pi$, so when the trajectory passes the minimum angle, θ_{min}, between $\boldsymbol{g}_{S}$ and $\boldsymbol{g}_{Des}$, then $\cos(\theta_{min})$ will be at a maximum. So, our goal here is to predict the time at which $\cos(\theta_{min})$ will be at a maximum and capture the image at that time. With $f(t) = \cos(\theta(t))$, then we have:

$$\dot{f} = \boldsymbol{g}_{Des}^{T}\dot{\boldsymbol{g}}_{S} \tag{106}$$

Recall the kinematic differential equation of the gravity vector in the sensor frame of reference:

$$\dot{\boldsymbol{g}}_{S}(t) = -[\boldsymbol{\omega}(t)]_{\times}\boldsymbol{g}_{s}(t) \tag{107}$$

We will assume for the prediction horizon that the angular velocity of the body is constant, a reasonable assumption since the prediction horizon is likely much less than one second. Making substitutions we have:

$$\dot{f} = -\boldsymbol{g}_{Des}^{T}[\boldsymbol{\omega}]_{\times}\boldsymbol{g}_{s}(t) \tag{108}$$

If we consider $t = 0$ the time at which we start looking for the minimum angle, and we know the gravity vector in the sensor frame at that time, $\boldsymbol{g}_{s}(0)$, then we may integrate the gyro to find $\boldsymbol{g}_{s}(t)$:

$$\dot{f} = -\boldsymbol{g}_{Des}^{T}[\boldsymbol{\omega}]_{\times}e^{-[\boldsymbol{\omega}]_{\times}t}\boldsymbol{g}_{S}(0) \tag{109}$$

Our goal now is to find the time at which $\dot{f} = 0$, which will correspond to the time when the camera is closest to the desired orientation. We start by applying Rodrigues' rotation formula to the matrix exponential

$$\dot{f} = -\boldsymbol{g}_{Des}^{T}[\boldsymbol{\omega}]_{\times}\left(I_3 - \frac{sin(\|\boldsymbol{\omega}\|t)}{\|\boldsymbol{\omega}\|t}[\boldsymbol{\omega}]_{\times}t + \frac{1 - cos(\|\boldsymbol{\omega}\|t)}{\|\boldsymbol{\omega}\|^2 t^2}[\boldsymbol{\omega}]_{\times}^{2}t^2\right)\boldsymbol{g}_S(0) \tag{110}$$

$$\dot{f} = -\boldsymbol{g}_{Des}^{T}\left([\boldsymbol{\omega}]_{\times} - \frac{sin(\|\boldsymbol{\omega}\|t)}{\|\boldsymbol{\omega}\|}[\omega]_{\times}^{2} + \frac{1 - cos(\|\boldsymbol{\omega}\|t)}{\|\boldsymbol{\omega}\|^2}[\boldsymbol{\omega}]_{\times}^{3}\right)\boldsymbol{g}_S(0) \tag{111}$$

For readability, let $A = \sin(\|\boldsymbol{\omega}\|t)$ and $B = 1 - \cos(\|\boldsymbol{\omega}\|t)$

$$\dot{f} = -\boldsymbol{g}_{Des}^{T}\left([\boldsymbol{\omega}]_{\times} - \frac{A}{\|\boldsymbol{\omega}\|}[\boldsymbol{\omega}]_{\times}^{2} + \frac{B}{\|\boldsymbol{\omega}\|^2}[\boldsymbol{\omega}]_{\times}^{3}\right)\boldsymbol{g}_S(0) \tag{112}$$

$$\dot{f} = -\boldsymbol{g}_{Des}^{T}[\boldsymbol{\omega}]_{\times}\boldsymbol{g}_S(0) + \left(\frac{\boldsymbol{g}_{Des}^{T}[\boldsymbol{\omega}]_{\times}^{2}\boldsymbol{g}_S(0)}{\|\boldsymbol{\omega}\|}\right)A - \left(\frac{\boldsymbol{g}_{Des}^{T}[\boldsymbol{\omega}]_{\times}^{3}\boldsymbol{g}_S(0)}{\|\boldsymbol{\omega}\|^2}\right)B \tag{113}$$

Again, to make things more readable, let:

$$k_1 = \boldsymbol{g}_{Des}^{T}[\boldsymbol{\omega}]_{\times}\boldsymbol{g}_S(0) \tag{114}$$

$$k_2 = \left(\frac{\boldsymbol{g}_{Des}^{T}[\boldsymbol{\omega}]_{\times}^{2}\boldsymbol{g}_S(0)}{\|\boldsymbol{\omega}\|}\right) \tag{115}$$

$$k_3 = \left(\frac{\boldsymbol{g}_{Des}^{T}[\boldsymbol{\omega}]_{\times}^{3}\boldsymbol{g}_S(0)}{\|\boldsymbol{\omega}\|^2}\right) \tag{116}$$

Then we have

$$\dot{f} = -k_1 + k_2 A - k_3 B \tag{117}$$

$$\dot{f} = -(k_1 + k_3) + k_2\, sin(\|\boldsymbol{\omega}\|t) + k_3\, cos(\|\boldsymbol{\omega}\|t) \tag{118}$$

We now combine the sin and cos terms into a single sinusoid using the Harmonic Addition Theorem [74]:

$$\dot{f} = -(k_1 + k_3) + \sqrt{k_2^2 + k_3^2}\, cos(\|\boldsymbol{\omega}\|t + \phi)\,,\ \ \phi = atan\left(-\frac{k_2}{k_3}\right) \tag{119}$$

Now set $\dot{f} = 0$ and solve for t:

$$t_{wait} = \frac{1}{\|\boldsymbol{\omega}\|}\left(acos\left(\frac{k_1 + k_3}{\sqrt{k_2^2 + k_3^2}}\right) - \phi\right) \tag{120}$$

Thus, once the FIM sensor detects it is within the θ_{Des} boundary, it will pause for t_{wait} before capturing the image.

Target attitude selection

The previous subsection presupposed that we had a target attitude ($\boldsymbol{g}_{Des}$) that would most likely have the plate of food in view. That is, we supposed there was some arm angle which, on average, contains the food. The following section describes a method to find such an attitude.

The general approach here is to experimentally find a target attitude by simultaneously tracking g_S while taking images. We can thus label images which contain food, and find some "mean" g_S from the labelled images. But, what is the meaning of a so-called "mean" g_S? That is, taking the mean of two norm constrained vectors does not produce a norm constrained vector! This is easily seen by considering a set of k norm-constrained vectors:

$$\boldsymbol{g}_i \in \{\boldsymbol{v} \in \mathbb{R}^3 | \boldsymbol{v}^T\boldsymbol{v} = \alpha,\ \alpha > 0\}, \qquad i = 1,2,\dots,k \tag{121}$$

Now, consider a proposed mean vector, $\overline{\boldsymbol{g}}$:

$$\overline{\boldsymbol{g}} = \frac{1}{k}\sum_{i=1}^{i=k}\boldsymbol{g}_i \tag{122}$$

Then, the squared-norm of $\overline{\boldsymbol{g}}$ is given by:

$$\overline{\boldsymbol{g}}^T\overline{\boldsymbol{g}} = \frac{1}{k^2}\left(\sum_{i=1}^{i=k}\boldsymbol{g}_i^T\right)\left(\sum_{i=1}^{i=k}\boldsymbol{g}_i\right) \tag{123}$$

$$\overline{\boldsymbol{g}}^T\overline{\boldsymbol{g}} = \frac{1}{k^2}\left(\sum_{i=1}^{i=k}\sum_{j=1}^{j=k}\boldsymbol{g}_i^T\boldsymbol{g}_j\right) \tag{124}$$

Note by definition of the inner product that:

$$\boldsymbol{g}_i^T\boldsymbol{g}_j = \alpha\cos\theta \tag{125}$$

where θ is the angle between $\boldsymbol{g}_i$ and $\boldsymbol{g}_j$, with $0 \leq \theta \leq \pi$. In the case where $\theta = 0$, then $\boldsymbol{g}_i = \boldsymbol{g}_j$, so we may write:

$$\boldsymbol{g}_i^T \boldsymbol{g}_j < \alpha, \qquad \boldsymbol{g}_i \neq \boldsymbol{g}_j \tag{126}$$

Substituting into the summation then gives:

$$\overline{\boldsymbol{g}}^T \overline{\boldsymbol{g}} < \frac{1}{k^2}(k^2 \alpha) \tag{127}$$

$$\overline{\boldsymbol{g}}^T \overline{\boldsymbol{g}} < \alpha, \tag{128}$$

Thus the mean vector will not belong to the norm-constrained set except in the trivial case where $\boldsymbol{g}_i = \boldsymbol{g}_j$ for all i and j.

So, what is the best way to estimate the target attitude, then? We could just normalize $\overline{\boldsymbol{g}}$, but is that optimal in any sense? To proceed, we can frame this as an optimization problem:

$$\begin{gathered} max \frac{1}{k} \sum_{i=1}^{k} \boldsymbol{g}_i^T \boldsymbol{x} \\ s.t.\ \boldsymbol{x}^T \boldsymbol{x} = \alpha \\ \boldsymbol{g}_i^T \boldsymbol{g}_i = \alpha, \qquad i = 1,2,..,k \end{gathered} \tag{129}$$

In other words, we seek a norm-constrained $\boldsymbol{x}$ which is on-average closest to the data set (the set of $\boldsymbol{g}_i$) in the sense of the inner product. The cost function is an affine function in $\boldsymbol{x}$, so it is convex, but the quadratic equality constraint is not convex. However, we can relax the constraint to an inequality. Consider some $\overline{\boldsymbol{x}}$ which maximizes the cost function with $\overline{\boldsymbol{x}}^T \overline{\boldsymbol{x}} < \alpha$. If we scale $\overline{\boldsymbol{x}}$ such that

$$\widetilde{\boldsymbol{x}} = \sqrt{\frac{\alpha}{\overline{\boldsymbol{x}}^T \overline{\boldsymbol{x}}}}\, \overline{\boldsymbol{x}}, \tag{130}$$

then we can write the following, since $\sqrt{\frac{\alpha}{\overline{\boldsymbol{x}}^T \overline{\boldsymbol{x}}}} > 1$:

$$\frac{1}{k} \sum_{i=1}^{k} \boldsymbol{g}_i^T \widetilde{\boldsymbol{x}} = \frac{1}{k} \sqrt{\frac{\alpha}{\overline{\boldsymbol{x}}^T \overline{\boldsymbol{x}}}} \sum_{i=1}^{k} \boldsymbol{g}_i^T \overline{\boldsymbol{x}} > \frac{1}{k} \sum_{i=1}^{k} \boldsymbol{g}_i^T \overline{\boldsymbol{x}} \tag{131}$$

Therefore, we can rewrite the optimization with an inequality constraint:

$$\begin{gathered} min -\frac{1}{k}\sum_{i=1}^{k} \boldsymbol{g}_i^T \boldsymbol{x} \\ s.t.\ \boldsymbol{x}^T \boldsymbol{x} - \alpha \leq 0 \\ \boldsymbol{g}_i^T \boldsymbol{g}_i = \alpha, \qquad i = 1,2,..,k \end{gathered} \tag{132}$$

which is a convex Quadratically Constrained Linear Program. Incidentally, the solution is easily found analytically. Rewrite the objective function:

$$\frac{1}{k}\sum_{i=1}^{k} \boldsymbol{g}_i^T \boldsymbol{x} = \bar{\boldsymbol{g}}^T \boldsymbol{x} = \|\bar{\boldsymbol{g}}\|_2 \|\boldsymbol{x}\|_2 \cos\theta \tag{133}$$

Clearly, the objective function is maximized when $\|\boldsymbol{x}\|_2 = \alpha$ (our constraint boundary) and $\theta = 0$, which maximizes the cosine term. In other words, the optimal solution is the unit vector in the same direction as $\bar{\boldsymbol{g}}$:

$$\boldsymbol{g}_{Des} = \frac{\sum_{i=1}^{k} \boldsymbol{g}_i}{\left\|\sum_{i=1}^{k} \boldsymbol{g}_i\right\|_2} \tag{134}$$

An important note here is that the absolute accuracy of this vector is not important because the system can easily be calibrated on a per-user basis. The calibration routine asks the user to wear the device and start eating food. The device will collect a series of images over the course of several bites. With each image, it will also save the estimate of the gravity vector when the image was taken. Then, a food detection neural network (or human) can identify which images contain food, and create a record of gravity vector estimates for which food was visible. Then a single target that is both device- and user-specific can be established using the previous discussion. The algorithm can be made more robust by having the user take off the device and put it back on, try different postures, eating styles, etc.

Convolutional Neural Network for food-containing image classification

For any ML algorithm, one of the main challenges is generating an adequate dataset, as it is often impractical to collect a large enough dataset for adequate training oneself. Fortunately there are several image datasets available online for us to leverage, and in this case we can utilize the CIFAR-100 dataset [75]. CIFAR-100 contains 60,000 images of size 32x32 pixels labeled according to 100 “fine” classes and 20 “coarse” classes. For instance, one of the coarse classes is “flower” with associated fine class “orchid.” For the FIM device, the relevant coarse class is “food containers” made up of objects including bottles, bowls, plates, and cups.

The goal of the FIM on-board classifier is to discriminate between an image containing food or not, a binary classification problem. Thus, the CIFAR-100 dataset was relabeled as "food container" or "not food container." When the dataset is labelled in this manner, it creates an imbalanced dataset with a ratio of 1:20 positive to negative examples. Training a classifier using a imbalanced dataset is likely to result in a classifier that simply outputs a negative class prediction for all images, so we rebalance the data set using under sampling by randomly selecting a subset of the negative class such that the training and test sets are equally balanced. Although more sophisticated resampling techniques are available [62], we obtained reasonable results using a basic random under sampling technique.

There are many approaches to image classification, but Convolutional Neural Networks (CNNs) remain the most popular network architectures (see [76] for an overview and history). In the context of image classification, CNNs are essentially a series of trainable image filters which operate on the image using a convolution operation. A visualization of a prototypical CNN is shown in Fig. 38. Intuitively, the filters are trained to learn important features such as object edges. Typically a down sampling operation called max pooling is used in between convolutional layers. The output of the convolutional and down sampling layers is then flattened into a vector, which is finally used as input into a multilayer perceptron network which provides the classification output. CNNs are typically very large in size. As an example, one popular and successful CNN for image classification, Alexnet, contains 60 million parameters and is eight layers deep (five convolutional layers, and three fully connected layers). Here, we follow a very similar CNN architecture to a recently reported CNN developed for embedded systems that achieved a respectable 86% classification accuracy using the CIFAR-10 dataset (a closely related image dataset that preceded CIFAR-100) [77]. The version of the CNN architecture for the network used in our work for food image classification is summarized in Table 10.

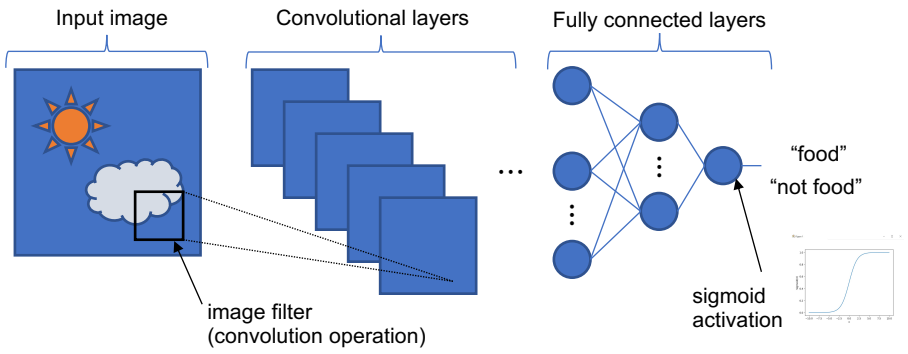

Fig. 38. Visualization of typical Convolutional Neural Network architecture.

TABLE 10: CONVOLUTIONAL NEURAL NETWORK FOR ON-BOARD FOOD IMAGE CLASSIFICATION

Layer type	Output shape	Number of parameters
input	32x32x1	0
convolution	30x30x32	320
max pooling	15x15x32	0
convolution	13x13x64	18496
max pooling	6x6x64	0
convolution	4x4x64	36928
flatten	1024	0
fully connected	64	65600
output	1	65

Results

Target orientation identification

In this section, experimental results for identifying the target orientation relative to gravity for optimal image capture of food are presented. The FIM device was secured to the wrist, and the user was seated in front of a plate of food and instructed to make the motions of eating. The camera was setup to continually capture images, and at each capture point, the estimate of gravity was also recorded. An example of the highest resolution color image the system can capture of a plate of food being eaten from the wrist position when the hand is near to the mouth is shown in Fig. 39. For algorithm development, the device camera was set to grayscale at the lowest resolution Quarter Common Intermediate Format (QCIF, 176x144 pixels) setting to reduce the data-transfer burden on the microcontroller. The low resolution camera setting is appropriate in the context of feeding the image into a classification network, which has an even lower resolution input (96x96 pixels). In practice, the high resolution image could be captured, downscaled for classification, and then the original transferred off the device once food is detected.

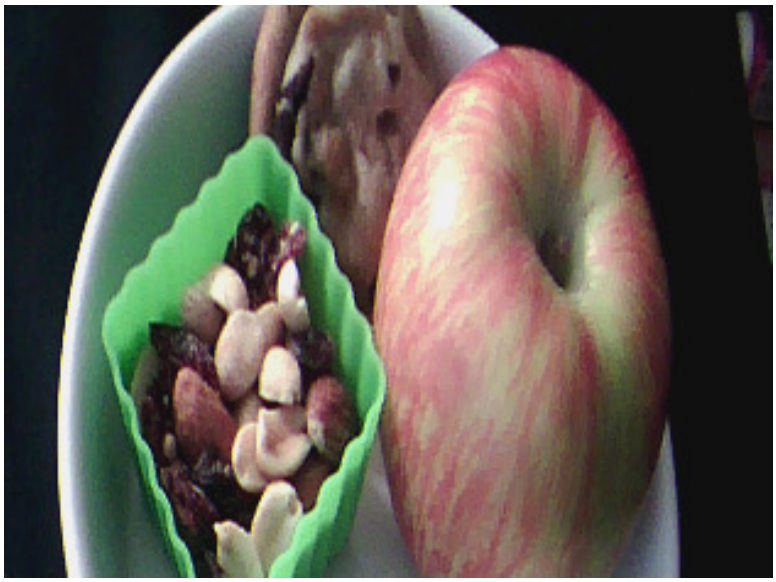

Fig. 39. High resolution FIM image of food
Example FIM image of a plate of food in color at a higher resolution setting, demonstrating the system is capable of capturing easily discernable images of the food.

A photo taken with a mobile phone from the view of the user is shown in Fig. 40 along with an example image taken from the FIM camera when worn on the wrist and set to the lower resolution. Although the FIM image is low resolution, a knife and slices of bread can be discerned. The camera lens used here has a fairly small field of view, which is desirable from a privacy perspective (though more challenging from an algorithm perspective).

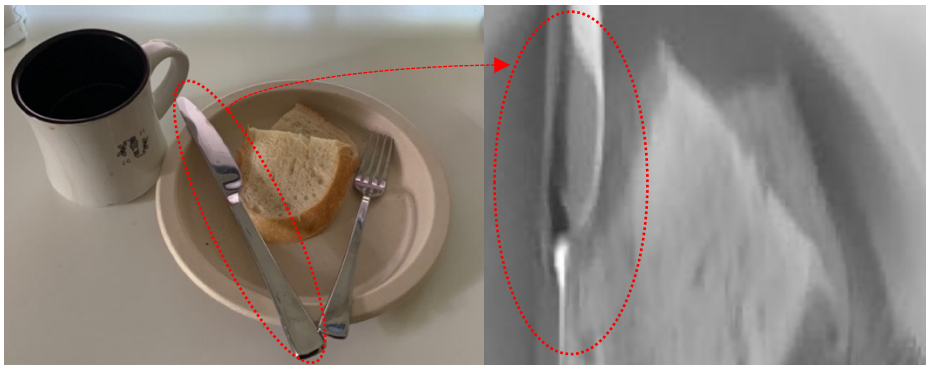

Fig. 40. Example images of the plate of food
Left image is taken with a mobile phone from the user's point of view. Right image is taken with the wrist-worn FIM device at the resolution used for classification. Even at very low resolution, the knife and bread slices are visible in the FIM image.

After the images were streamed to a host computer, they were processed and manually labelled with metadata according to whether or not food was visible in the image and the estimated gravity vector when the image was taken. A plot of the gravity vector estimate of 133

images taken while repeatedly moving the hand from the plate to the mouth is shown in Fig. 41. Although the gravity vector contains only two degrees of freedom, it is intuitive to view all three x, y, and z elements on a unit sphere. As shown in the figure the images that contain views of the food were generally clustered in the middle portion of the arm motion.

The labelled images were then used to find a target vector for taking an image as described previously. The data is shown in 2D in Fig. 42 along with the estimated target vector.

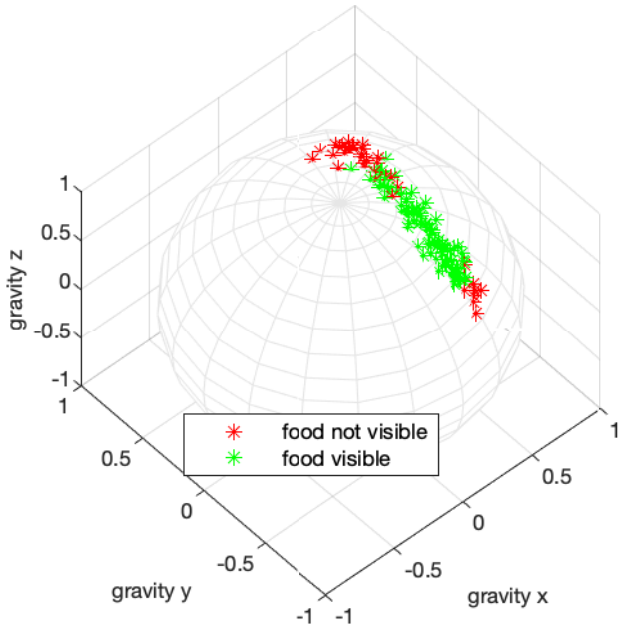

Fig. 41. Gravity vector and food visibility
Experimental results showing the recorded gravity vector estimate when an image was recorded. Images were coded according to whether food was visible in the image. Food was generally visible in the middle section of the eating trajectory.

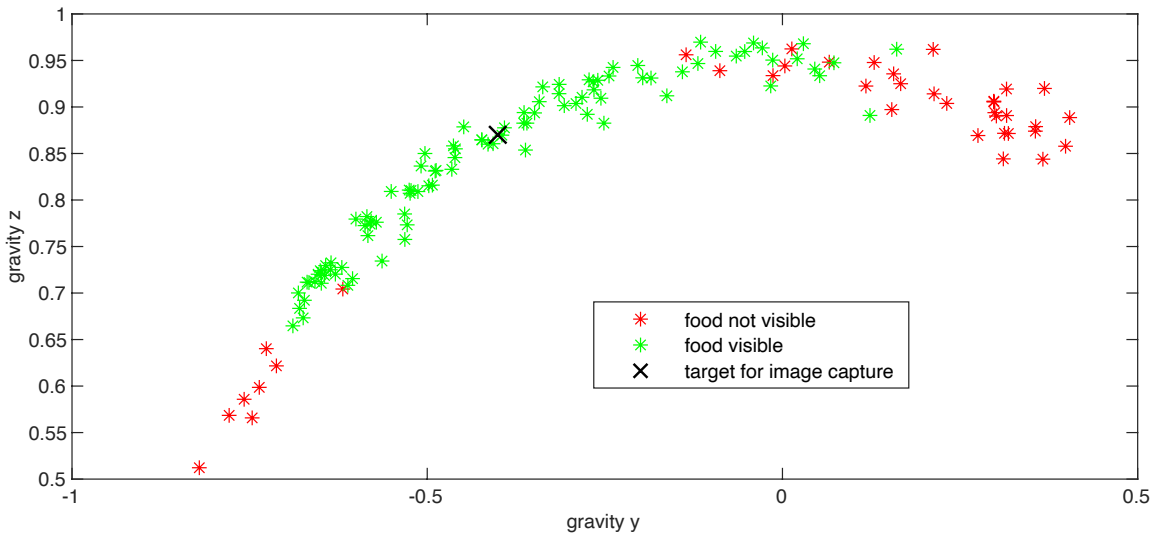

Fig. 42. Estimated food image capture target
Experimental results showing the recorded gravity vector estimate when an image was recorded along with the estimated target direction for image capture.

A point of interest in the data set is that the subjective "best view" of the food was not centered amongst the set of all food-visible data points. To show this, the quality of the image was subjectively scored on a scale of 0 to 1, with a score of 0 indicating the food was not visible at all, and a score of 1 indicating the entire image frame contained food. In Fig. 43, we plot the image gravity vectors colored according to three categories (food not visible, food partially visible, and food fully visible), and the best views of the food are notably biased towards one side of the band of all images with food at least partially visible. This result suggests that the approach of finding the so-called "mean" gravity vector may be flawed. Perhaps we should instead center the target on the band of images ranked 1. However, centering the target on the best images may increase the risk of missing the food entirely, since the cluster of best images neighbors a cluster of images with zero visibility of the food! Thus, the better approach could be to bias the target towards the best images but not fully centered on them.

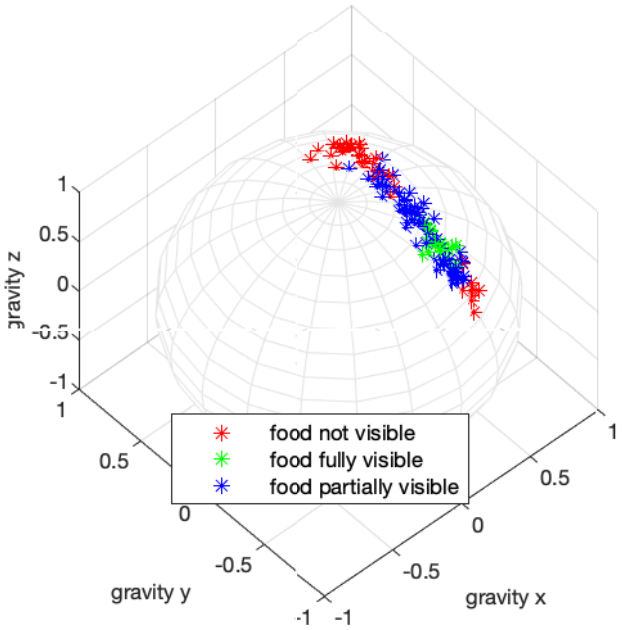

Fig. 43. Image quality ranking of food visibility
Experimental results showing the recorded gravity vector estimate when an image was recorded. Images were coded according to whether food was fully visible in the image, partially visible, or not visible at all. The best images are biased towards one end of the band of all food-visible images.

To introduce such a bias, we can use a tuning heuristic in the form of a weighting parameter during target selection:

$$\boldsymbol{g}_{Des} = \frac{\sum_{i=1}^{k} f(s_i, \beta)\boldsymbol{g}_i}{\left\|\sum_{i=1}^{k} f(s_i, \beta)\boldsymbol{g}_i\right\|_2} \tag{135}$$

where $f(s_i, \beta)$ is some weighting function of the score, s_i, of image i, and β is a tunable parameter to adjust how strongly the target is biased towards the best images. As we wish to always use the best images during target selection and discard the images containing no visible food, a candidate for the weighting function is a monomial of order β:

$$f(s, \beta) = s^{\beta}, \qquad \beta > 0 \tag{136}$$

Thus, for very small β, the target will be selected using all images containing partial or full views of the food weighted nearly equally. As β increases, the target will increasingly be biased towards the best views, with very large β mostly ignoring the partial-view images. An alternative to using the weighting function is to use a threshold during target selection. For instance, we might decide to use only images with score greater than 0.75 when selecting the target capture direction. However, this could potentially introduce unintended discrete "jumps" in the target direction. The

weighting function is a continuous function of all images containing at least partial views of food, and so it can be tuned smoothly. An example procedure of subjectively tuning the target would be to start with β close to zero and increase it until the image quality improves without introducing an undesired number of images containing no food. An example of finding the capture target using a weighted procedure is shown in Fig. 44. In this case, the mean score of the nearest 25 images to the target was 0.88 for the unweighted target and 0.93 for the weighted target.

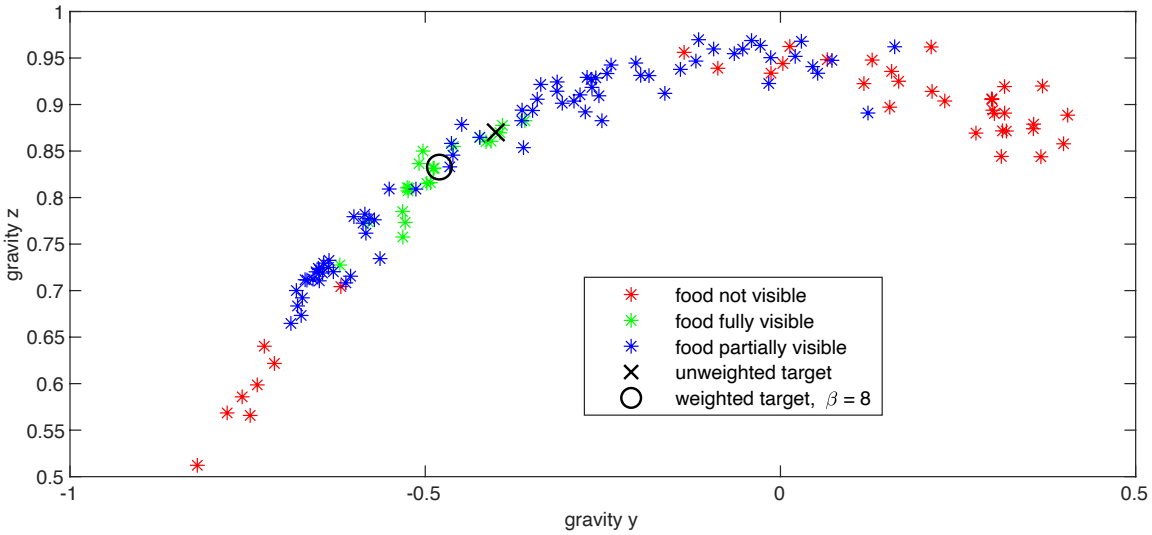

Fig. 44. Weighted image capture target selection
Experimental results showing the recorded gravity vector estimate when an image was recorded. The resulting target direction using a weighted and unweighted selection is shown. The weighted target is biased towards the best images but not fully centered on them.

We now consider if the results obtained in the previous experiment are generalizable. Specifically, a second set of experiments was run in which the FIM device was taken off and put back on. The sensor orientation on the wrist was approximately the same as before, but no careful alignment or calibration procedure was performed. Next, instead of eating using the fingers, the second set of data contains the motions of eating with a fork. A new plate of food was placed in front of the user. The new plate was put in approximately the same position as before, but no careful alignment or fiducial markings were used. As before, the FIM camera was set to continually capture images while recording the gravity vector estimate at the time of image capture. A total of 129 images were taken while eating the rice. An image of the second plate of food taken with a mobile phone from the user's perspective along with an image taken from the FIM device is shown in Fig. 45.

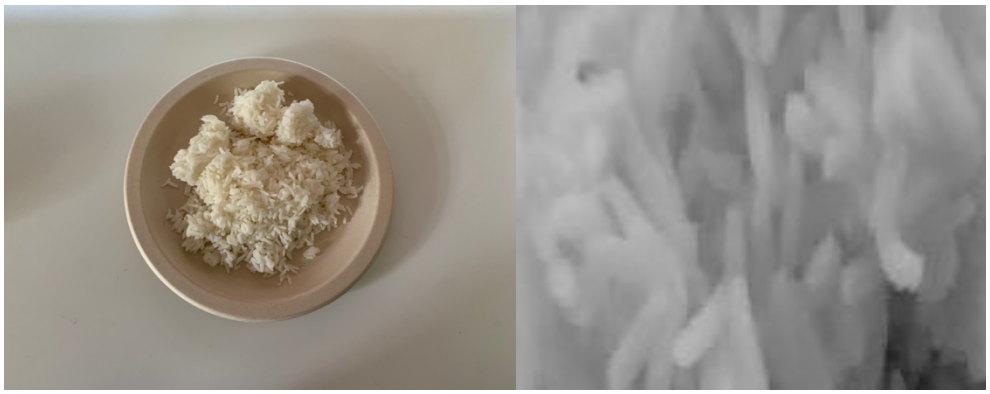

Fig. 45. Example images of the second plate of food
Left image is taken with a mobile phone from the user's point of view. Right image is taken with the wrist-worn FIM device. The rice is visible in the FIM image.

In Fig. 46, the estimated gravity vector for each image taken while eating the rice are overlayed with the previous gravity vector estimates while eating with the fingers. In this figure, it can be seen that the trajectories overlap for much of the motion, suggesting the plate of rice could be in view at similar estimates of the gravity vector.

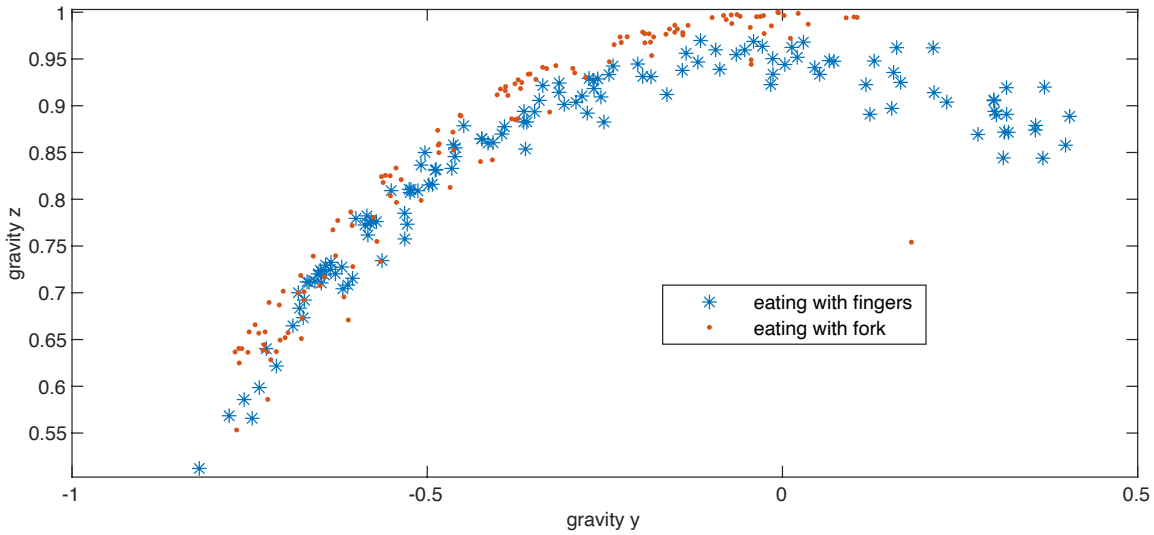

Fig. 46. Gravity vector while eating with a fork and finger
Gravity vector y and z component estimates recorded for each image during eating with the fingers compared to eating with a fork. The trajectories overlap for much of the motion.

Next, we find the closest five images to our gravity target selected using the weighting function from the first data set. The five closest images are found by forming a matrix containing the gravity vector estimates from the new image set and premultiplying by the target. The five

closest images in the angular sense will be the five greatest elements of the resulting proximity vector p:

$$\boldsymbol{p} = \boldsymbol{g}_{Des}^{T}[\boldsymbol{g}_1 \quad \cdots \quad \boldsymbol{g}_k] \tag{137}$$

The resulting closest five images in the new data set are shown in Fig. 47. Despite the motion blur (unavoidable with the available hardware setup), four of the five closest images do contain rice in the entire image.

Fig. 47. The five closest images taken while eating with a fork to the target gravity vector
The target was selected using the first data set containing eating with the fingers. Four of the five images contain full views of the rice.

Due to hardware limitations with the current camera setup, it is difficult to experimentally implement the wait-time prediction algorithm. As the images tend to get blurry except for the slowest motions, the prediction wait times tend to be on the order of several seconds. A prediction of this length is likely inaccurate because (a) the motion is likely to change in that time span, and (b) the gyroscope quality is not high enough for this length of prediction. However, we can demonstrate the general principle using simulation. In Fig. 48, a simulated trajectory is shown along with two image capture bounds. When the camera is triggered by the loose threshold (15°), an image is taken, but because the threshold is so large, the target image will be off center. When the camera is triggered by the tight threshold (5°), no image is taken, as the trajectory never gets close enough to the target. When the camera is triggered using the gyroscope prediction by waiting approximately 500 ms, the resulting image is at the best possible orientation relative to the target given the predicted trajectory. More work is needed in this area to show the predictive timing improves results experimentally.

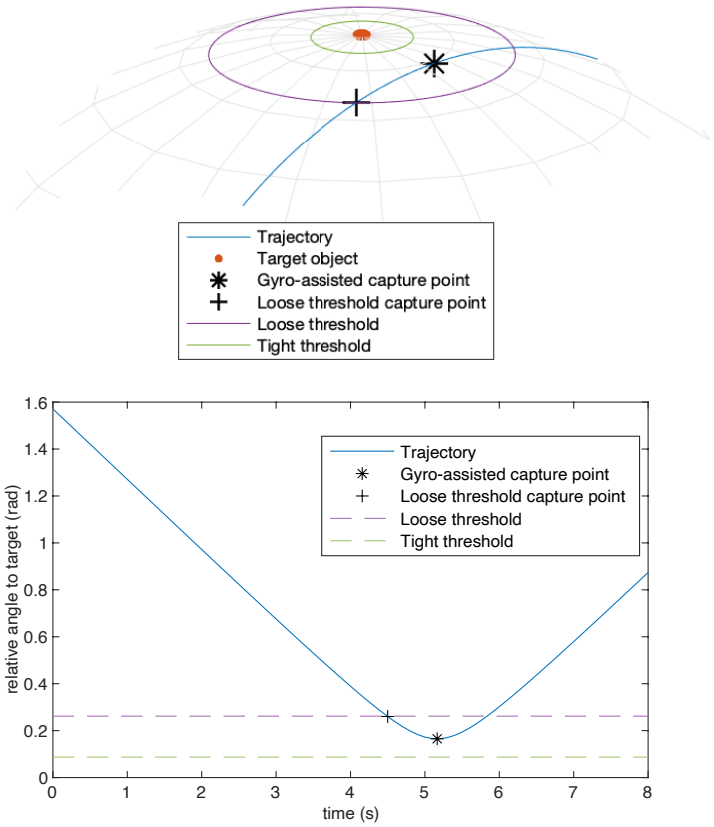

Fig. 48. Gyro-assisted image capture
Simulation (3D view top, 2D view bottom) of a camera trajectory comparing image capturing triggered by an angle threshold (cross marker) and image capture triggered by gyro-assisted motion estimation (asterisk marker). When a loose threshold (15 deg.) is utilized, images are always captured on the threshold boundary and may be far away from the object. With a tight threshold (5 deg. as shown), the trajectory may miss the threshold entirely, missing the opportunity to capture even an off-centered image of the object. The gyro-assisted image capture allows the device to predict when the camera will have the best possible view of the object and to take an image at that point in time.

Image classification

The food image CNN was trained by using the CIFAR-100 training set as the training images, and the independent CIFAR-100 test set as a validation set. After training, the CNN achieved a 94% accuracy on the training set and an 81% accuracy on the test set. The confusion matrix for the test set is shown in Fig. 49. The training was completed using 32-bit floating point parameters, so the network was then quantized post-training to 8-bit integers to further decrease the required memory of the network and enable microcontroller deployment and edge inference. The loss in accuracy on the test set between pre- and post-quantization was 0.1%. After quantization, the CNN requires only 129 KB of memory, which easily fits onto the 1MB flash of the Arduino Nano BLE used in this work. As a reference, Alexnet image classifier discussed above requires 233 MB to store its parameters, although direct comparisons are not appropriate,

as Alexnet is a multi-class classifier with different objectives. Example images correctly classified by the quantized CNN on both the CIFAR-100 test set and using images from the actual FIM camera are shown in Fig. 50.

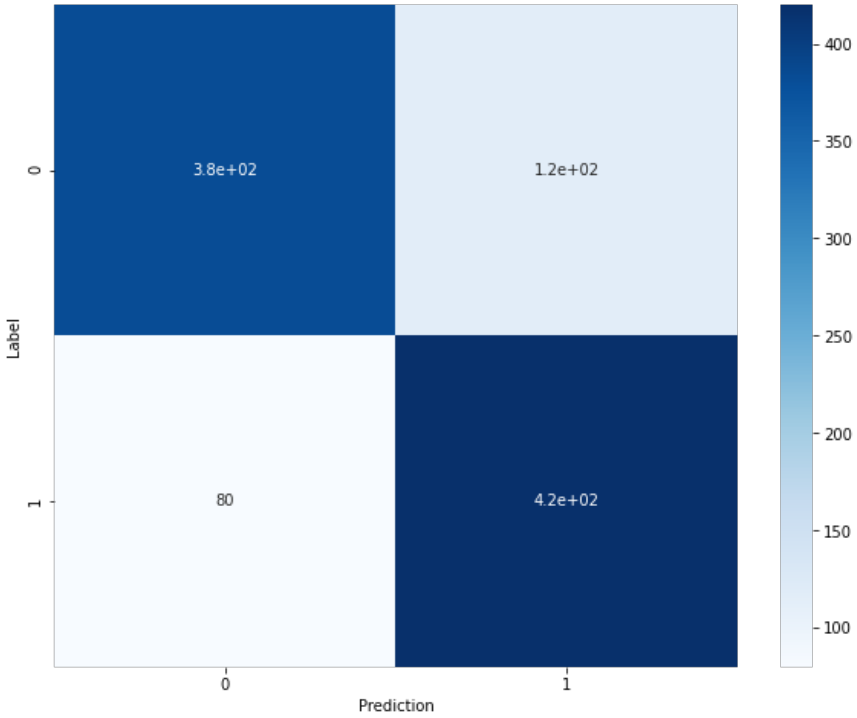

Fig. 49. CNN confusion matrix on the test set
Label 0 corresponds to not food, and label 1 corresponds to food.

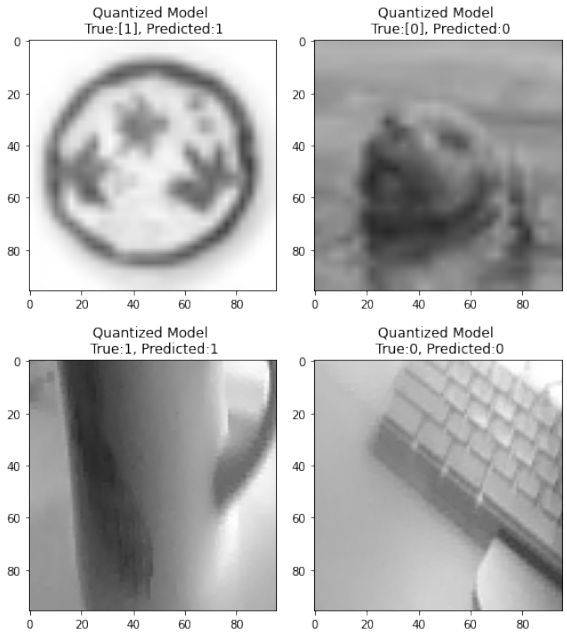

Fig. 50. Example classifications made by the CNN
The top row contains images from the CIFAR test image set. The CNN correctly classifies the plate (top left) as a food object and the dog (top right) as not a food object. The bottom row contains images from the actual FIM camera, showing correct prediction of a coffee mug (bottom left) and a keyboard (bottom right).

Conclusion

In this chapter, we have extended the initial work on the FIM device and developed methods to automatically capture images of the food being consumed once eating has been detected. Specifically, the FIM device estimates the device orientation relative to gravity and uses the IMU to optimally time the taking of the image. It then submits a downsampled version of the image to a lightweight convolutional neural network specially quantized for on-device classification to increase the likelihood that the captured image indeed contains food. The CNN achieved an overall accuracy of 81% on an independent test set. State of the art multi-class classifiers developed by Google have achieved over 95% accuracy on CIFAR-100 dataset [78], but we emphasize that the CNN developed in this chapter takes only 129 KB of memory and can easily be deployed onto a microcontroller for edge inference. One benefit of the edge inference approach is that the device can immediately reattempt taking a photo if the image does not contain food without having to spend the power and time to wirelessly transfer the image and wait for a cloud server to respond. Such an approach further increases the overall privacy of the

device and reduces the power needed to wirelessly transfer data. Future work may involve further improving the classifier and implementing hardware improvements allowing faster camera frame rates.

Chapter 7:
Estimation of mobile device position inside moving vehicles using IMUs

Introduction

Distracted driving from phone use is a major cause of vehicular accidents in the United States. The National Safety Council recently estimated that 26% of vehicular accidents involve phone usage or texting while driving [79], and recent research estimated that nearly half of teen drivers engaged in texting while driving [80]. This research also found that phone usage while driving was correlated with other risky driving behavior such as not wearing a seat belt and driving under the influence of alcohol. Thus, there is a clear need to discourage phone use while driving, and one approach to the problem is to automatically disable certain phone features (such as texting) while the user is actively driving. Automatic phone disabling requires the phone to first decide if it is in a moving vehicle and to second decide if it is in the driver seat position. The first decision is relatively straightforward based on GPS speed data, while the second decision concerning phone location inside the vehicle is more challenging.

Previous research has been published reporting methods to detect if the phone is in the driver's position [81]–[86]. Wang and colleagues [81] report a left/right localization method by comparing the phone's accelerometer data to a reference accelerometer placed in the center of the vehicle (a small accessory plugged into the cigarette lighter port). Based on the measured centripetal acceleration during a turn, the system can determine if the phone is on the passenger or driver side of the reference, though no method is presented to differentiate between front and back. In [82], the authors use a very similar centripetal acceleration approach, but two phones are required, which must be carefully synchronized to within 100ms of each other. This requirement limits the practical application of such a system. Front and back position of the phone is determined by analyzing the signature pattern of road bumps on accelerometer data. An entirely different approach in [83] utilizes acoustic ranging to localize the phone inside the car by sending timed high frequency beeps from the car's audio system speakers. This approach requires a

Bluetooth connection to the vehicle. Bo et al. [84] employed a machine learning approach to determine which side of the car the user enters and then use acceleration data during road bump events to determine if the phone is in the front or the rear. The authors also attempt to identify abnormal texting behavior (i.e. fast, intermittent typing may indicate texting while driving). The authors report an 87% recognition accuracy using a Naive Bayes classifier, though it is unclear how generalizable the algorithm is. Chu et al. [85] also use a machine learning approach, and train several support vector machines to recognize micro-motions related to driving, such as pressing the car pedal and entering the vehicle on the left hand side. The overall accuracy is 85%, but the approach fails depending on how the phone is carried, for instance by a woman in a purse.

The current generation Apple iPhone can detect when the user is in a moving car and will automatically turn off notifications. Patent filings assigned to Apple indicate that Apple has proposed to detect if the phone is being used by the driver based on a simple speed threshold and additional image processing from the phone's camera [86]. However, the phone's user may have privacy concerns with granting camera access to the phone while driving.

Thus, none of the systems in the literature provide a reliable system that only utilizes the sensors already on the smartphone. The most practical and ideal detection scheme of the phone's location inside the moving vehicle would have the following characteristics:

- The system should require no external hardware beyond the accelerometer, gyroscope and GPS sensors already available on the phone itself.
- The system should require no wireless connection to any other existing hardware on the vehicle, such as through Bluetooth.
- The system should require no special instrumentation of the car at all.

This chapter will present a novel algorithm for determining the location of a smartphone inside a moving vehicle using only the motion signals from phone's three-axis gyroscope, accelerometer and GPS. This method relies on an understanding of the vehicle's pitch and roll dynamics [87]. As will be shown through simulation and experiments, when the car decelerates, the nose of the car will pitch downward, rotating about the vehicle pitch center. Thus, the front seat positions experience a downward motion while the rear seat positions experience an upward motion. The opposite is true while accelerating. Likewise, while turning, the vehicle tends to roll towards the outside of the turn about the roll center. While turning left, the left seat positions experience an upward motion, and the right seat positions experience a downward motion. The opposite is true during a right turn.

In combination, the pitch and roll dynamics offer the ability to uniquely identify seat position relative to the pitch and roll centers via motion signals recorded by the phone's IMU, in particular the accelerometer and gyroscope. Once these signals are transformed into the vehicle's coordinate frame, features extracted in real-time from a moving window of motion data can be used for classification of the phone's location. This paper utilizes a linear support vector machine (SVM) to classify these features and identify if the phone is in the driver, the front passenger, the rear left passenger, or rear right passenger position. The feasibility of this approach is presented first through simulation and then through extensive experimental data.

Compared to previous research and available commercial technology, the main contributions of this paper are as follows:

1. The derivation of a method to identify seat position inside a moving vehicle based on the pitch and roll dynamics of a vehicle. Real-time data and knowledge of vehicle dynamics is exploited for the first time for this task in literature.
2. The development of a new method to transform the motion signals recorded in the phone's frame of reference to the vehicle's frame of reference.
3. The development of a completely self-contained system that only uses sensors already available on the phone.

Simulation and experimental demonstrations of the feasibility to classify the phone's seat position inside a moving vehicle using a machine learning algorithm based only on features extracted from a real-time moving window of motion data.

The remainder of this chapter is organized as follows. In Methods, the pitch and roll dynamics of a moving vehicle are presented along with a method to determine the orientation of the phone relative to the vehicle, necessary to transform the measured motion signals from the phone to the vehicle's coordinate frame of reference. Simulation studies are also presented. In Results, experimental findings are presented demonstrating a smartphone IMU is capable of measuring the signals necessary for seat classification. Finally, seat position classification using a linear support vector machine is presented.

Methods

Vehicle dynamic model

The classification algorithm developed in this paper relies on the pitch and roll dynamic behaviors of the vehicle [87]. Half car models for pitch and roll dynamics for preliminary simulation evaluations are presented in this section. The coordinate system for the car is defined

as forward longitudinal ($+x$), left lateral ($+y$), and up vertical($+z$). The rotational angles are roll ϕ about the x axis and pitch θ about the y axis.

The free body diagram for the vehicle roll model is shown in Fig. 51. Here, the vehicle rotates about the roll center due to the lateral acceleration caused by turning and by lateral translation. The suspension force is modeled as a linear spring and damper with deflection $z(t) = \left(\frac{\ell_w}{2}\right)\sin(\phi)$, spring constant k, and damping coefficient b as follows:

follows:

$$F_S(t) = k\left(\frac{\ell_w}{2}\right)sin(\phi) + b\left(\frac{\ell_w}{2}\right)\dot{\phi}cos(\phi) \tag{138}$$

where F_S is the suspension force, ϕ is the roll angle, ℓ_w is the track width of the vehicle, k is the suspension stiffness and b is the damping coefficient.

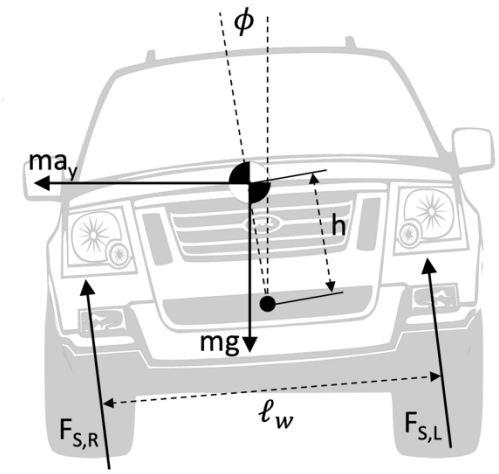

Fig. 51. Free body diagram for vehicle roll.

The lateral acceleration is treated as a force input at the center of gravity (CG) in the moment balance equation taken about the roll center. This couples the roll dynamics to the lateral dynamics through the lateral acceleration a_y term:

$$\begin{aligned}(I_{xx} + mh^2)\ddot{\phi} = ma_y hcos(\phi) + mghsin\,(\phi)\\ -k\left(\frac{\ell_w^2}{2}\right)sin(\phi) - b\left(\frac{\ell_w^2}{2}\right)\dot{\phi}cos(\phi)\end{aligned} \tag{139}$$

where $(I_{xx} + mh^2)$ is the moment of inertia taken at the roll center, h is the distance from the roll center to the CG, m is the vehicle mass and g is the gravitational constant. More detailed simulation models for roll dynamics can be found in [88]–[90]. However, the model presented here is adequate for the roll dynamics in order to evaluate in simulations how the lateral motion

and acceleration of the car cause roll and how these signals influence the accelerometer and gyroscope signals read by the sensors on the phone. Extensive experimental evaluations of the developed algorithm using measurements on real car maneuvers will later be presented to serve as a more reliable indicator of the developed algorithm's effective performance.

A similar equation results for the vehicle pitch dynamics if the variables in (138) and (139) are redefined according to Fig. 52 with the appropriate substitutions of I_{yy} for I_{xx} and the longitudinal acceleration a_x for a_y. In this case, the equation relating the pitch dynamics to the longitudinal acceleration and deceleration is given by:

$$\begin{aligned}&\left(I_{yy}+mh^2\right)\ddot{\theta}=ma_xhcos(\theta)+mghsin(\theta)\\&-k\left(\frac{\ell_f^2+\ell_r^2}{2}\right)sin(\theta)-b\left(\frac{\ell_f^2+\ell_r^2}{2}\right)\dot{\theta}cos(\theta)\end{aligned}\qquad(140)$$

where, θ is the pitch angle, $\left(I_{yy}+mh^2\right)$ is the moment of inertia taken at the pitch center, h is the height from the pitch center to the CG, m is the vehicle mass, k is the suspension stiffness, b is the damping coefficient, ℓ_f is the distance from the CG to the front tires, ℓ_r is the distance from the CG to the rear tires and g is the gravitational constant.

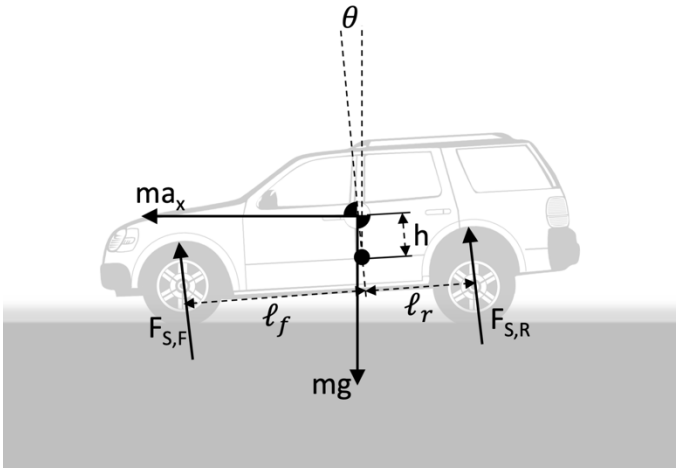

Fig. 52. Free body diagram for vehicle pitch.

Real-time estimation of phone-to-vehicle orientation

The smartphone user may carry the phone in his/her pocket, in a purse, on a display stand in the car or they may be holding it in their hands. Thus the orientation of the phone with respect to the car can be completely arbitrary and all three Euler angles of rotation can be unknown. Since the motion signals recorded by the phone must be transformed from the phone coordinate frame

to the vehicle coordinate frame, real-time knowledge of the orientation of the phone relative to the vehicle is required.

There are several possible approaches to finding the orientation of the phone, two of which are presented next, and their relative advantages and limitations discussed. Please note that orientation can be parameterized a number of ways, the three primary methods being Euler angles, quaternions, or the 3x3 direction cosine matrix (DCM). A complete discussion of the merits and drawbacks of these three parameterizations is outside the scope of this paper, but the DCM (also known as the rotation matrix) is used here, mainly because it avoids singularity issues of other parameterizations. A full treatment of orientation parameterizations may be found in [5].

One potential method to estimate the phone orientation relative to the car is to first estimate the orientation of the phone relative to earth, and then the orientation of the vehicle relative to earth. Call these orientation DCMs (or rotation matrices) R_{earth}^{phone} and R_{earth}^{car}, respectively, where R_a^b refers to the DCM between frames a and b. Once these are known, the orientation of the phone relative to the car can be determined as:

$$R_{car}^{phone} = (R_{earth}^{car})^T R_{earth}^{phone} \tag{141}$$

R_{earth}^{car} is relatively straightforward to estimate using GPS course information available on the phone, which reports the direction of travel of the vehicle. A typical method to estimate R_{earth}^{phone} would be to utilize measurements of two known inertial directions, such as the gravity vector with the accelerometer and earth's magnetic field, which is locally a constant vector, with the magnetometer. These can then be optimally fused with gyroscope measurements using, for instance, a Kalman Filter. The main problem with this approach is that the presence of unknown magnetic disturbances, such as steel components in the car, can badly corrupt the magnetic measurement.

Here, a simpler approach is used to directly determine R_{car}^{phone} without needing to transform to the earth frame first. At any given time, the accelerometer measurement is the vector sum of gravity and the vehicle acceleration:

$$\boldsymbol{a}_{measured} = \boldsymbol{a}_{gravity} + \boldsymbol{a}_{vehicle} \tag{142}$$

While braking, the dominant components of the measured acceleration will be gravity plus the longitudinal deceleration of the vehicle, with smaller components being road vibrations and the roll/pitch transient dynamics. Thus, while braking, the measurement can be approximated as:

$$\boldsymbol{a}_{measured} = \boldsymbol{a}_{gravity} + \boldsymbol{a}_{longitudinal} \tag{143}$$

While stopped (as determined by GPS) or by averaging acceleration data over a long period of time, a good measurement of $a_{gravity}$ is available. This can then be subtracted from the total measured acceleration during a brake maneuver to determine the longitudinal acceleration. Thus, the gravity measurement gives the direction of the vehicle z-axis, while the longitudinal acceleration during braking gives the vehicle x-axis. From these, R_{car}^{phone} can be estimated using a static attitude determination algorithm such as TRIAD [5], presented next.

Let $\{v_1, v_2, v_3\}$ be the representation of an orthonormal set of vectors in the vehicle's frame, and let the representation of these same vectors in the phone's frame be denoted as $\{p_1, p_2, p_3\}$. Since R_{car}^{phone} is the rotation matrix that defines the orientation of the phone to the car, then:

$$R_{car}^{phone} = [\boldsymbol{p}_1 \quad \boldsymbol{p}_2 \quad \boldsymbol{p}_3][\boldsymbol{v}_1 \quad \boldsymbol{v}_2 \quad \boldsymbol{v}_3]^T \tag{144}$$

It is assumed that the terrain is flat, i.e. there is no significant road gradient. Let $\boldsymbol{r}_{g,c} = [0 \quad 0 \quad -1]^T$ represent the gravity vector in the vehicle frame and $\boldsymbol{r}_{g,p}$ represent a measurement of the gravity vector in the phone frame. Likewise, let $\boldsymbol{r}_{x,c} = [1 \quad 0 \quad 0]^T$ represent the longitudinal axis of the vehicle and $\boldsymbol{r}_{x,p}$ be a measurement of the vehicle's longitudinal direction in the phone's frame obtained during a braking maneuver, as explained above. R_{car}^{phone} can now be determined from (144) by creating the following sets of orthonormal reference and body vectors:

$$\boldsymbol{v}_1 = \boldsymbol{r}_{g,c} \quad , \quad \boldsymbol{v}_2 = \frac{\boldsymbol{v}_1 \times \boldsymbol{r}_{x,c}}{\|\boldsymbol{v}_1 \times \boldsymbol{r}_{x,c}\|} \quad and \quad \boldsymbol{v}_3 = \boldsymbol{v}_2 \times \boldsymbol{v}_1 \tag{145}$$

$$\boldsymbol{p}_1 = \frac{\boldsymbol{r}_{g,p}}{\|\boldsymbol{r}_{g,p}\|} \quad , \quad \boldsymbol{p}_2 = \frac{\boldsymbol{p}_1 \times \boldsymbol{r}_{x,p}}{\|\boldsymbol{p}_1 \times \boldsymbol{r}_{x,p}\|} \quad and \quad \boldsymbol{p}_3 = \boldsymbol{p}_2 \times \boldsymbol{p}_1 \tag{146}$$

Experimental verification of phone orientation estimation

To demonstrate the use of the above method, accelerometer and gyroscope data from a smartphone (Apple iPhone 6s running Sensor Play application) was collected at 100Hz during normal driving with the phone in three different orientations, as shown in Fig. 53.

In Fig. 53A, the orientation of the phone relative to the car was visually aligned such that the phone's y-axis was aligned with the vehicle's x-axis, and the phone's z-axis was vertical. The vector $\boldsymbol{r}_{g,p} = [0.04 \quad 0.02 \quad -0.99]^T$ was determined by taking the mean accelerometer measurement when the vehicle speed measurement was 0. The vector $\boldsymbol{r}_{x,p} = [0.0 \quad 0.30 \quad 0.00]^T$ was determined by subtracting $\boldsymbol{r}_{g,p}$ from the mean accelerometer

measurement during the brake maneuver at 12s. Equation (144) gives a very reasonable estimate of the phone orientation as:

$$R_{car}^{phone} = \begin{bmatrix} 0.02 & 1.00 & 0.02 \\ -1.00 & 0.02 & -0.05 \\ -0.05 & -0.02 & 1.00 \end{bmatrix} \tag{147}$$

The above approach to estimating the phone orientation was repeated on two additional orientations, shown in Fig. 53B and Fig. 53C with results summarized in Table 11. In experiment B, the phone was placed horizontal with the display up on the center console, rotated 135 degrees clockwise about the vertical axis, as measured with a 45-degree triangle referenced against the vehicle console surface. In experiment C, the phone was placed in the vehicle cup holder at a vertical angle of 45 degrees, measured with an inclinometer application on the phone. The +x axis of the phone was visually oriented with the -y axis of the vehicle. As seen from a comparison of the third and fourth columns of Table 11, in all three experiments, the method provides very good estimates of the phone orientation relative to the vehicle. It should be noted that the small differences between the entries in columns 3 and 4 are likely due to the phone not exactly being at 45^{o} orientations in case B and case C, since these orientations were only done visually or approximately.

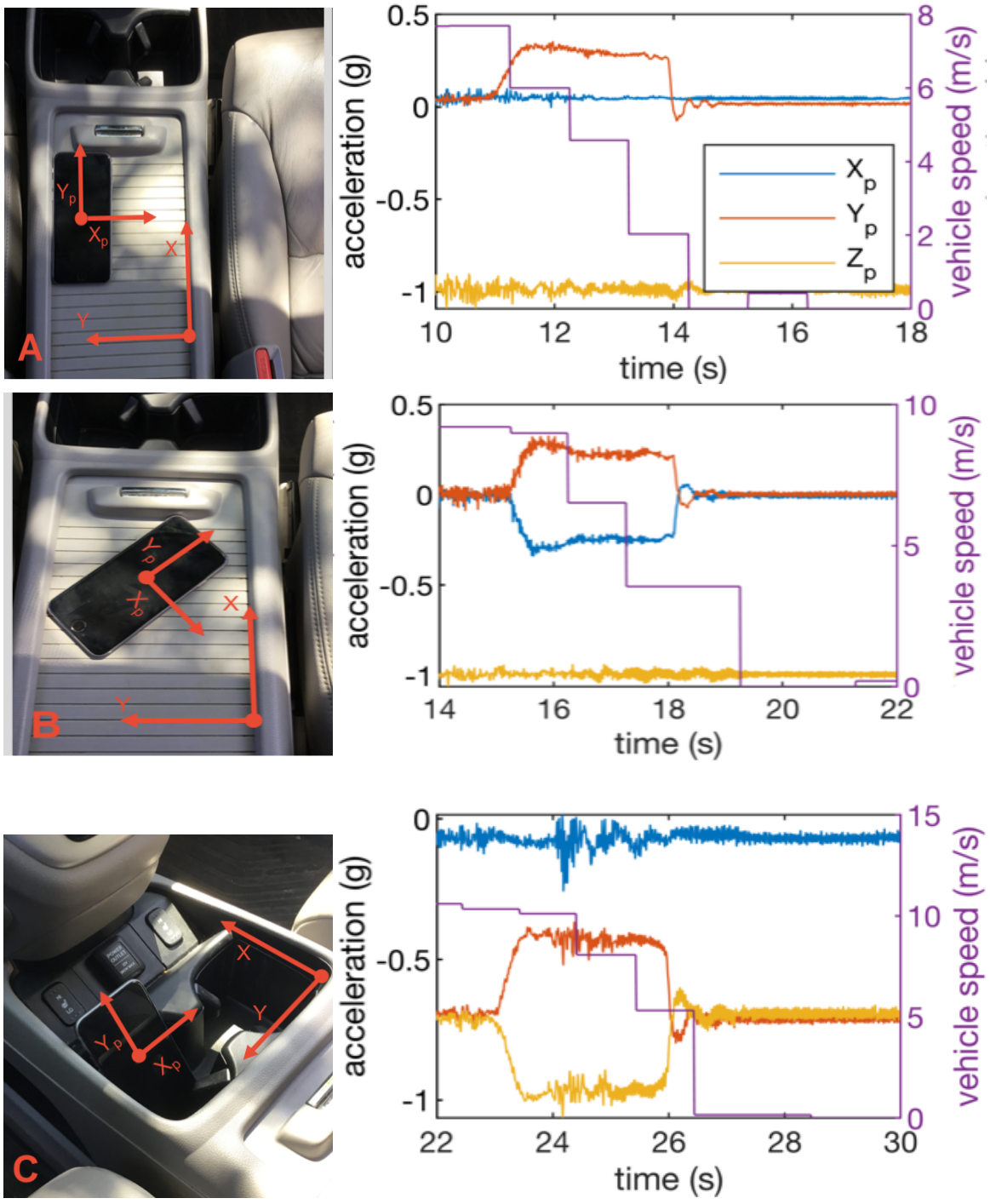

Fig. 53. Three experiments to demonstrate the phone orientation estimation
Experiments demonstrating the procedure to estimate phone orientation using the measured acceleration during braking events as a reference for the vehicle longitudinal axis. The event can be detected as braking by the decrease in speed, available from the phone's GPS data (purple line in the data plots).

TABLE 11: EXAMPLE DCM ESTIMATES FOR THREE PHONE ORIENTATIONS

ID	$\boldsymbol{r}_{g,p}$	$\boldsymbol{r}_{x,p}$	R_{car}^{phone} (experimental)	R_{car}^{phone} (expected)
A	$\begin{bmatrix} 0.04 \\ 0.02 \\ -0.99 \end{bmatrix}$	$\begin{bmatrix} 0.00 \\ 0.30 \\ 0.00 \end{bmatrix}$	$\begin{bmatrix} 0.02 & 1.00 & 0.02 \\ -1.00 & 0.02 & -0.05 \\ -0.05 & -0.02 & 1.00 \end{bmatrix}$	$\begin{bmatrix} 0.0 & 1.00 & 0.0 \\ -1.00 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.00 \end{bmatrix}$
B	$\begin{bmatrix} 0.00 \\ 0.00 \\ -1.00 \end{bmatrix}$	$\begin{bmatrix} -0.24 \\ 0.22 \\ 0.00 \end{bmatrix}$	$\begin{bmatrix} -0.74 & 0.67 & 0.00 \\ -0.67 & -0.74 & 0.00 \\ 0.00 & 0.00 & 1.00 \end{bmatrix}$	$\begin{bmatrix} -0.707 & 0.707 & 0.00 \\ -0.707 & -0.707 & 0.00 \\ 0.00 & 0.00 & 1.00 \end{bmatrix}$
C	$\begin{bmatrix} -0.07 \\ -0.71 \\ -0.70 \end{bmatrix}$	$\begin{bmatrix} 0.01 \\ 0.28 \\ -0.28 \end{bmatrix}$	$\begin{bmatrix} -0.03 & 0.70 & -0.71 \\ -1.00 & 0.03 & 0.07 \\ 0.07 & 0.71 & 0.70 \end{bmatrix}$	$\begin{bmatrix} 0.0 & 0.707 & -0.707 \\ -1.00 & 0.0 & 0.0 \\ 0.0 & 0.707 & 0.707 \end{bmatrix}$

Simulation studies

The objective of this section is to evaluate the pitch angular velocity and vertical acceleration responses of the vehicle during braking and the roll angular velocity and vertical accelerations during turning. The nature of these responses, as measured by the sensors on the smartphone, will depend on the location of the phone inside the car. As we shall see, the nature of these responses can serve as a reliable indicator of the phone location and whether it is located in the driver seat of the car.

Fig. 54 shows the simulation response of the car to a series of step changes in longitudinal acceleration. At $t = 1s$, the brakes and a longitudinal deceleration of $0.4g$ are applied until $t = 5s$. At $t = 8s$, an acceleration of $0.4g$ is applied until $t = 12s$. The top part of Fig. 54 shows the vertical acceleration and the pitch angular rate at the front seats of the car. The bottom part of Fig. 54 shows the vertical acceleration and the pitch rate at the rear seats of the car. The step response shows an underdamped oscillatory behavior in both sensor signals. Clearly, the vertical accelerations in the front and back are equal but opposite in direction. On the other hand, the pitch angular rate (as measured by a gyroscope) is the same in both the front and rear seats of the car during braking/ acceleration.

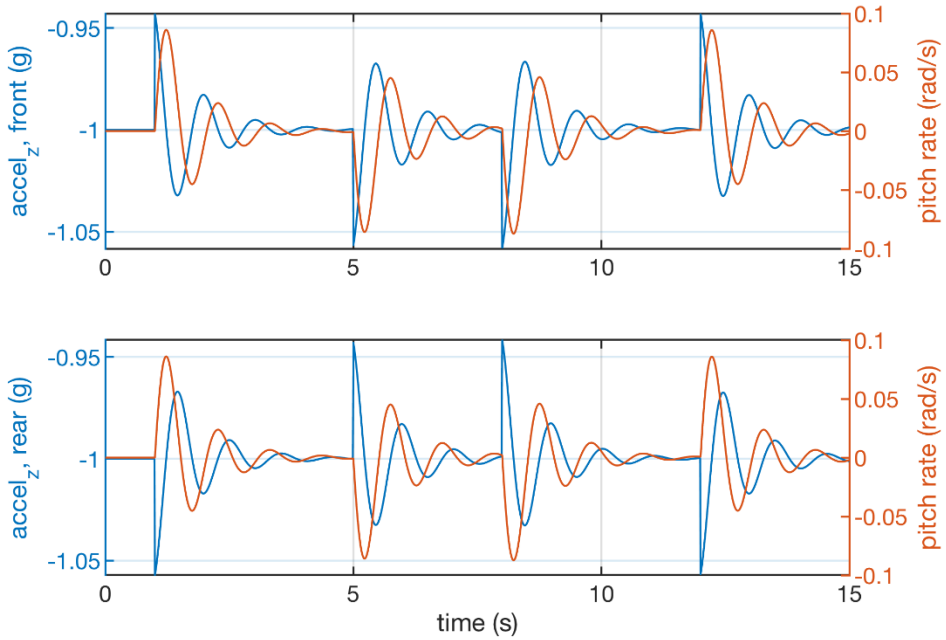

Fig. 54. Simulated response to several step changes in longitudinal acceleration
At t=1s, the brakes and a longitudinal deceleration of 0.4g are applied until t=5s. At t=8s, an acceleration of 0.4g is applied until t=12s. The top plot shows the pitch rate and the vertical acceleration for the front seats, and the bottom plot shows the pitch rate and the vertical acceleration for the rear seats.

In order to capture the differences in the vertical acceleration behavior at the front and back seats, integrating the vertical acceleration twice shows the rear vertical position is positive when the pitch angle is positive, and the opposite behavior for the front vertical position. This suggests a useful feature for front/back classification would be the correlation of the vertical-axis accelerometer data integrated twice with the pitch-axis gyroscope data integrated once during periods with longitudinal acceleration, as summarized in Table 12 below. Likewise, a useful feature for left/right classification would be the correlation of the vertical-axis accelerometer data integrated twice with the roll-axis gyroscope data integrated once during a turn. However, integrating accelerometer signals twice and gyroscope signals once is non-trivial due to the presence of bias errors in the sensor signals. These bias errors are inevitable and time-varying so that they will always exist and cannot be compensated fully. Careful high and low pass filtering could solve some of these bias and integration-drift issues, but nonetheless, trying to experimentally calculate vertical displacements and comparing to pitch and roll angles is not an easy approach.

TABLE 12: RELATIONSHIPS BETWEEN PITCH AND ROLL ANGLES VS. VERTICAL DISPLACEMENT			
Position	a_x	$\phi_{pitch} = \int g_y$	$z = \iint a_z$
Front	> 0 (accelerating)	< 0	> 0
	< 0 (braking)	> 0	< 0
Rear	> 0	< 0	< 0
	< 0	> 0	> 0
Position	a_y	$\phi_{roll} = \int g_x$	$z = \iint a_z$
Left	> 0 (left turn)	> 0	> 0
	< 0 (right turn)	< 0	< 0
Right	> 0	> 0	< 0
	< 0	< 0	> 0

A more desirable approach is to find classification features in the accelerometer and gyroscope signals that don't require integration at all. Careful inspection again of the data in Fig. 54 shows that the oscillatory responses for front and back acceleration are 180-degrees out of phase. Further, when comparing the acceleration responses to the pitch rate at each location, it is clear that there is a phase lag between these signals and the phase lag is different at the front and rear seats of the car. This phase lag could therefore be a useful classification feature. Hence, the signal cross covariance between these two sensor signals is utilized, in which the signal means are subtracted from two signals and the cross correlation is calculated as follows [91]:

$$c_{xy} = \begin{cases} \sum_{n=0}^{N-m-1} (x_{n+m} - \bar{x})(y_n - \bar{y}), & m \geq 0 \\ c_{xy}(-m), & m < 0 \end{cases} \tag{148}$$

where x is the first sensor signal (from the accelerometer), y is the second sensor signal (from the gyroscope) and m is the size of the lag, in terms of number of samples, which is being used to calculate the cross-correlation.

The cross covariances between vertical acceleration and pitch-axis angular rate are shown in Fig. 55 for the simulation data of Fig. 54, for a range of lag values. It can be seen that the cross-covariance decreases with time lag for the sensor signals on the front seats of the car. On the other hand, the cross-covariance increases with time lag for the sensor signals in the rear seats of

the car. The clear difference between the cross covariances for front and rear and their different behavior with time lag suggest that this could be a useful classification feature that can be reliably used to detect phone location. The same signal-processing approach can also be exploited with the roll gyroscope signal and vertical accelerometer signal to determine location being on the left or right side of the car. This simulation section has suggested a signal-processing approach for front-back and right-left classification of the phone location. In the following section, this signal-processing approach is applied to experimental data and its performance in a number of experimental car maneuvers is evaluated.

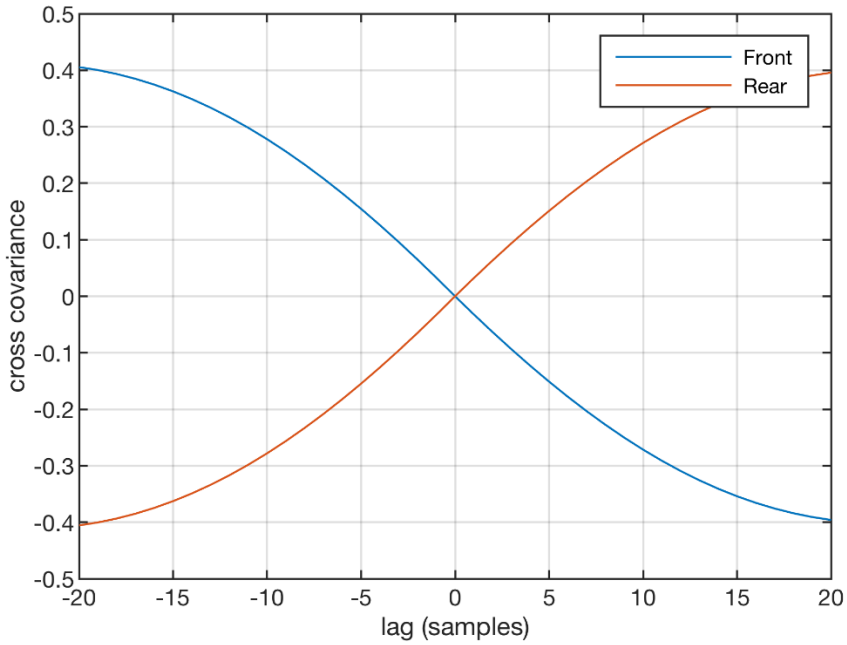

Fig. 55. Cross covariance of vertical acceleration to pitch-axis angular rate of simulated data.

Results

In this section, experiments are performed to verify whether the cross-covariance between the vertical accelerometer and the pitch or roll gyroscope signals can reliably identify the location of the smartphone in the car. In particular, the cross-covariance with the pitch gyroscope and its variation with time lag is used to identify whether the phone is in the front or rear seats of the car. The cross-covariance with the roll gyroscope and its variation with time lag is used to identify whether the phone is in the left or right sides of the car.

A smartphone (Apple iPhone 6S) was used to collect data in a car while being used at different seat locations in the car. A sampling rate of 100 Hz was used and the data collected was

entirely from the sensors only on the phone. The driving took place alongside normal day time traffic on urban roads and the speeds varied but were kept to less than 35 mph (16 m/s).

The phone was placed in each of four positions (driver, front passenger, back left passenger, back right passenger) for two data sets of 5 minutes each for all four seat positions. Thus a total of 40 minutes of data was obtained. To keep the phone from shifting, it was secured to the vehicle carpet using hook and loop strap. For the front seat positions, the phone was secured to the vehicle carpet approximately under the driver's left or front passenger's right knee, thus midway between the dash and the front seat in the longitudinal direction. For the rear seat positions, the phone was secured to the vehicle carpet where the back of the seat attaches to the vehicle frame, roughly approximating the left pants pocket position of the left rear passenger and right pants pocket of the right rear passenger. An example section of data is shown in Fig. 56. Zoomed in sections of experimental data are shown in Fig. 57 for front and rear positions while braking and in Fig. 58 for left and right positions while turning. For the pitching oscillations near 2.5 Hz in Fig. 57, the phase difference between the pitch-axis angular rate and vertical acceleration is clearly evident. As expected, the two signals have a phase difference of 90 degrees, while the sign of the accelerometer is opposite when comparing front to rear seat positions. For a 2.5 Hz oscillation, a 90 degree phase difference suggests the cross covariances should have a peak or valley near a lag of 10 samples for a 100 Hz sampling frequency.

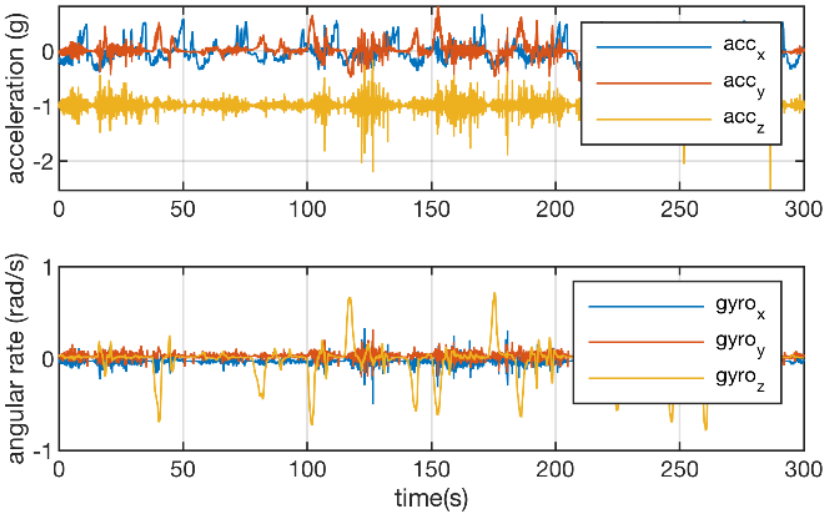

Fig. 56. Accelerometer and gyroscope data taken during city driving for back right passenger.

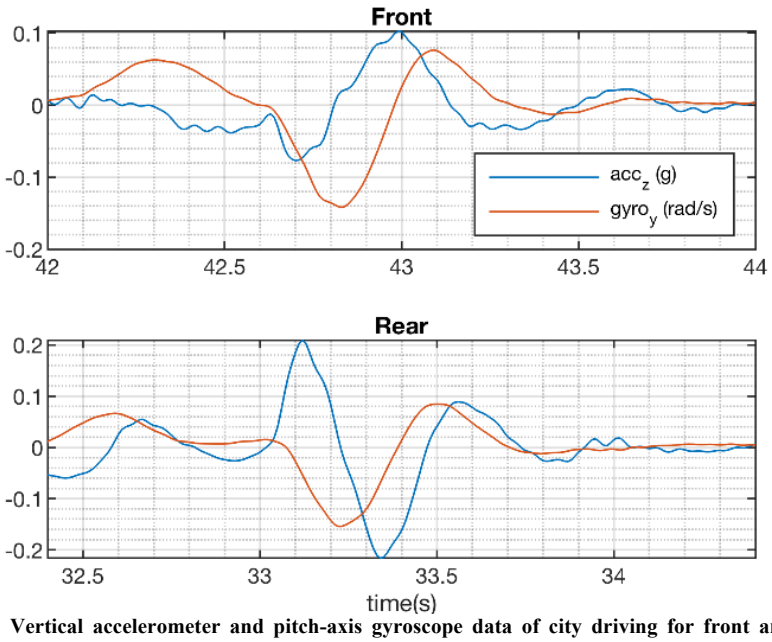

Fig. 57. Vertical accelerometer and pitch-axis gyroscope data of city driving for front and rear positions

Example data with gravity removed from the accelerometer signal. The signals are from two different braking events at the moment the vehicle comes to a complete stop and returns to a horizontal position. The 90 degree phase difference between the pitch axis gyroscope and vertical accelerometer signals is evident, while the accelerometer signal has the opposite sign comparing front to rear.

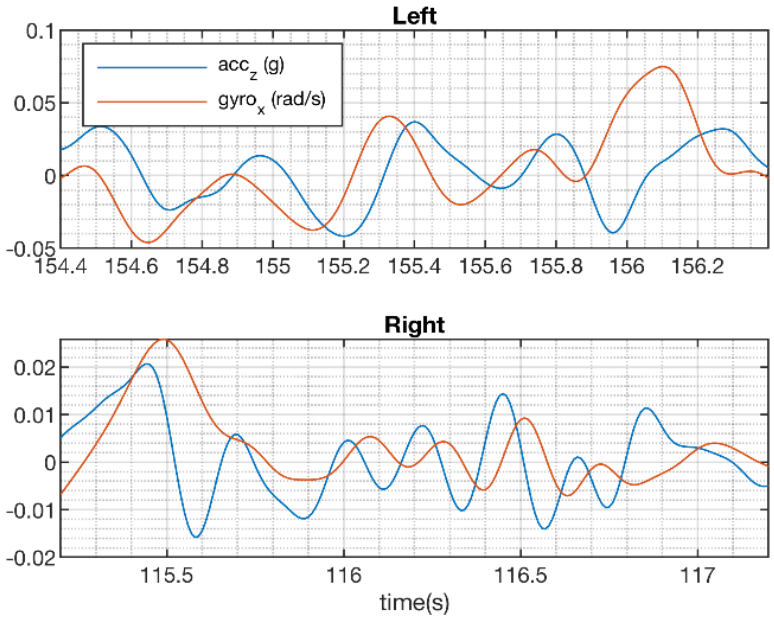

Fig. 58. Vertical accelerometer and roll axis gyroscope data of city driving for left and right positions

Example data with gravity removed from the accelerometer signal. The signals are from two different turning events. The 90 degree phase difference between roll axis gyroscope and vertical accelerometer signals is evident, while the accelerometer signal has opposite sign comparing left to right.

The cross covariances for vertical acceleration to pitch-axis angular velocity and vertical acceleration to roll-axis angular velocity are shown in Fig. 59 for a five minute data set of each seat position. The raw data was first low pass filtered prior to computing the cross covariances. Fig. 59 clearly shows that each seat position has a unique combination of cross covariances. As expected, the cross covariances have peaks and valleys near a lag of 10 samples. The experimental data follows the same overall behavior as the simulated cross covariances in Fig. 55. This indicates that the cross covariance is a good choice as a classification feature.

After experimental verification of the basic behavior of cross-covariance of the accelerometer-gyroscope pairs with phone location inside the car, a systematic algorithm for automatically detecting phone location is developed in the next section.

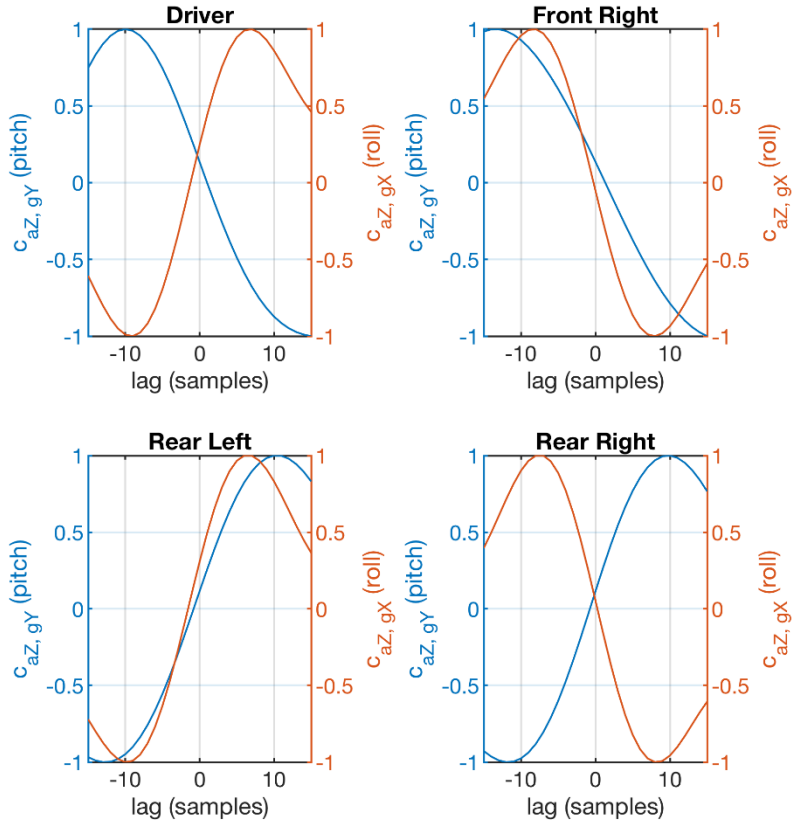

Fig. 59. Normalized cross covariances for experimental driving data
Each data set represents 5 minutes of driving. Blue curves are cross covariance of vertical acceleration and pitch-axis angular rate. Red curves are cross covariance of vertical acceleration and roll-axis angular rate. Each seat position has a clearly unique pattern of cross-covariances and its variation with time lag.

Phone localization using a support vector machine

The experimental results of the previous section verified that there is a clear and reliable relationship between phone location and the cross-covariance of accelerometer-gyroscope paired data. The next step is to develop and implement an automatic location identification algorithm that makes use of these cross-covariances. This section develops a machine learning algorithm consisting of a support vector machine for performing this automatic location identification.

Fig. 59 shows cross-covariances of experimental data for the 4 different locations of the phone. In each set of data, the cross-covariances between pitch gyroscope and vertical accelerometer and between roll gyroscope and vertical accelerometer are shown. In each data set 5 minutes of driving data is used. It can be seen that the slopes of the two cross-covariances are unique in the 4 different phone locations. For example, in the driver seat location, the pitch gyroscope cross-covariance has a negative slope while the roll gyroscope cross-covariance has a

positive slope around the zero lag point. Likewise, there is a unique combination of positive and negative slopes for the 4 different locations.

Ideally, the phone should recognize its location inside the vehicle in a much shorter period than 5 minutes, so the same data was divided up into shorter 45 second long sections, overlapping by 15 seconds. For each window, the two cross covariances were calculated, and a linear slope was fit to the cross-covariance curves in the $lag = \pm 5$ region. Fig. 60 shows the cross-covariance slopes for each of these 45 second samples. The plot shows good feature separation for use in a linear classifier.

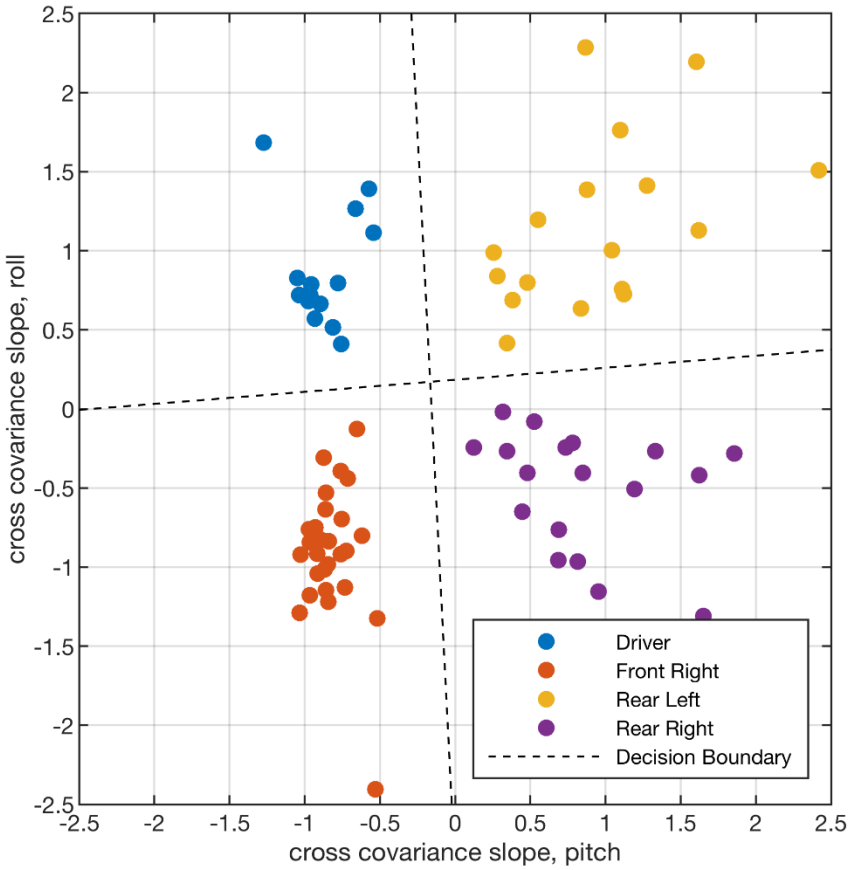

Fig. 60. Feature data used to train the SVM classifier
Data has been normalized to have zero mean and standard deviation of one. Each marker represents a 45-second-long experimental data set of city driving. The x-axis is the fitted slope of the cross covariance of vertical acceleration and pitch-axis angular velocity. The y-axis is the fitted slope of the cross covariance of vertical acceleration and roll-axis angular velocity. The dashed lines represent the SVM decision boundaries.

The above features were used as training inputs to a multi-class classifier made up of two binary support vector machines (SVM) [67]. The SVM fits the maximum margin hyperplane in the feature space by maximizing the distance between the hyperplane and the nearest data points.

In the case of a 2-dimensional feature space, the hyperplane is a line. The data here consisted of 77 labelled 45 second samples of driving data. The SVM was trained using all the data and validated using k-fold cross validation, which is preferred over a simple random separation of the data into single train and single test sets [68]. In k-fold cross validation, the 77 samples are randomly divided into k groups, or folds. One fold is held out, and the remaining k-1 folds are used to train an SVM. This process is then repeated a total of k times, such that each group gets used once as the test group. The overall accuracy of the SVM trained using all of the data is estimated by taking the mean accuracy of the k SVMs. Common choices for k are 5 and 10. As k gets larger, the computational cost increases, as more models must be trained, but the estimated performance approaches the true performance [68]. The limit for k is n, where n is the number of labelled data samples. In this case, the validation scheme is referred to as leave-one-out cross validation, as only one sample is held out to test, and n SVMs are trained.

Here, the accuracy of the classification for the vehicle experimental data was found to be 100% under both 5-fold and 10-fold cross validation, meaning that each binary SVM (left/right and front/rear) predicted the held-out fold with 100% accuracy for all 5 and 10 training rounds, respectively. Finally, experimental validation was also performed using leave-one-out cross validation, and the classification accuracy was again found to be 100% on all 77 folds. The dashed lines in Fig. 60 are the maximum margin hyperplanes for the two binary SVMs.

Conclusions

This chapter presented a novel method for the localization of a smartphone inside a vehicle using the motion data gathered by the IMU in the phone. Unlike previous work, the method requires no external hardware, no wireless communication with the vehicle, and no access to the phone's camera. It only uses the sensors already available on the phone.

First, the orientation of the phone relative to the car is determined, which can be done without relying on the error-prone magnetometer. Based on the vehicle dynamics measured by the phone, it was shown that using the cross covariances of vertical acceleration to the pitch-axis angular velocity and vertical acceleration to the roll-axis angular velocity provides a viable classification feature for determining which seat the phone is in. A total of 40 minutes of driving data was collected and used to train two binary support vector machines to differentiate between front/rear and left/right seat positions. The support vector machines achieved 100% accuracy on the 40 minute data set, validated using 5-fold, 10-fold, and leave-one-out cross validation. The 100% accuracy with experimental data shows that the developed system works extremely reliably.

Distracted driving due to phone usage is a major cause of vehicular accidents in the US, and there is a clear need for algorithms that can detect when a phone user is driving and automatically disable distracting features. Current phone apps are only able to detect whether the phone is in a moving car and then disable the use of texting during car motion. However, these apps are unable to detect whether the phone is being used by the driver or merely by a passenger of the car. The technology developed in this paper could thus be a very valuable tool for correctly disabling texting features only for the driver of the vehicle.

A limitation of the technology developed in this paper is that the orientation of the phone is assumed to be constant. If the orientation varies during use, then the developed signal processing algorithms will still work, but only if the change in orientation is slow, so that the orientation computation algorithm can recompute the lateral and longitudinal acceleration correctly. It is also assumed that the terrain is flat. Significant road gradients can therefore result in errors.

Chapter 8:

Conclusions

The proliferation of wearable smart devices in recent years has opened up many new opportunities in improving human health. Step trackers are commonly used to promote and increase daily physical activity, but there are many opportunities for utilizing the commodity sensors available on most wearables for more advanced and sophisticated health applications that have yet to be addressed. In particular, the availability of low-cost inertial measurement units (IMUs) provide the basis for applications involving motion analysis on the human body, particularly for applications involving at-home, out-of-clinic, and/or general-population use. Mobile-phone-grade IMUs suffer from several fundamental issues related to bias instability, and motion analysis that relies on the geomagnetic field is prone to unavoidable issues related to disturbances from magnetic fields inside buildings that can easily dwarf the geomagnetic field. This thesis has made contributions to the wearable health device arena through the development and experimental application of three health related applications that overcome these fundamental issues in the areas of respiratory analysis, diet monitoring, and vehicular safety.

Monitoring the physical motions of breathing is critical for many diseases, ranging from neuromuscular disorders to chronic obstructive pulmonary disease. We have developed a novel IMU-based respiratory monitoring device that has a combination of capabilities and portability unavailable with existing technologies on the market. The system consists of multiple IMUs attached to the thoracoabdominal surface, and respiratory displacements at points on the surface are estimated through careful removal of gravity from the acceleration signal, double integration, and filtering. The displacement accuracy is on the same order as a gold-standard optical motion tracking system. From displacements, a full suite of clinically-relevant temporal, phasic, and volumetric respiratory variables may be estimated. After initial development utilizing single-

subject experiments, the device is being used in a small-cohort IRB-approved study. We presented initial findings from the IRB study.

The second application area involves dietary monitoring and developing novel technology to improve the current methodologies in obesity research which have a notoriously poor track record in providing a complete and accurate record of dietary intake. This thesis has made progress in improving diet tracking methodologies through the development of a new food intake monitoring device. The device consists primarily of a wrist-worn IMU which uses motion signals to determine when the user's hand is near the mouth. The device then classifies such activity as eating or not, and in cases where eating is detected, it attempts to capture an image of the food using an on-board camera. An algorithm was developed for optimizing the timing of image capture based on partial orientation estimates provided by a Kalman filter and orientation prediction using the gyroscope. Once the image is taken, a convolutional neural network adapted for deployment on the device's microcontroller classifies the image as containing food or not.

Finally, this thesis developed a novel algorithm concerning vehicular safety due to distracted use of wearable and other personal devices such as smartphones while driving. Distracted driving due to device usage is known to be one of the major causes of vehicular accidents, and one possible methodology in reducing such accidents is for the device to automatically detect when the user is driving and automatically disable itself. Towards this end, we have developed an algorithm which localizes the device inside the vehicle without requiring any extraneous hardware or communication with the vehicle. By using the device's IMU to capture signals related to the dynamics of the vehicle, the algorithm first determines the orientation of the device relative to the vehicle by identifying the vertical and longitudinal axes of the vehicle. It then uses the cross correlation of the vertical acceleration and rotation rates of the pitch and roll axes as feature inputs to a support vector machine to determine which seat position the device is in. Both simulation and experimental results are used to develop the algorithm and demonstrate its successful application in real-world driving scenarios.

Bibliography

[1] "Above Avalon: Apple Watch Is Now Worn on 100 Million Wrists," *Above Avalon*. https://www.aboveavalon.com/notes/2021/2/11/apple-watch-is-now-worn-on-100-million-wrists (accessed Jun. 24, 2021).

[2] T. Harris *et al.*, "Physical activity levels in adults and older adults 3–4 years after pedometer-based walking interventions: Long-term follow-up of participants from two randomised controlled trials in UK primary care," *PLOS Medicine*, vol. 15, no. 3, p. e1002526, Mar. 2018, doi: 10.1371/journal.pmed.1002526.

[3] K. R. Foster and J. Torous, "The Opportunity and Obstacles for Smartwatches and Wearable Sensors," *IEEE Pulse*, vol. 10, no. 1, pp. 22–25, Jan. 2019, doi: 10.1109/MPULS.2018.2885832.

[4] M. V. Perez *et al.*, "Large-Scale Assessment of a Smartwatch to Identify Atrial Fibrillation," *New England Journal of Medicine*, vol. 381, no. 20, pp. 1909–1917, Nov. 2019, doi: 10.1056/NEJMoa1901183.

[5] F. L. Markley and J. L. Crassidis, *Fundamentals of Spacecraft Attitude Determination and Control*, vol. 33. New York: Springer, Springer New York, 2014. doi: 10.1007/978-1-4939-0802-8.

[6] H. Chao, C. Coopmans, L. Di, and Y. Chen, "A comparative evaluation of low-cost IMUs for unmanned autonomous systems," in *2010 IEEE Conference on Multisensor Fusion and Integration*, Sep. 2010, pp. 211–216. doi: 10.1109/MFI.2010.5604460.

[7] W. H. K. de Vries, H. E. J. Veeger, C. T. M. Baten, and F. C. T. van der Helm, "Magnetic distortion in motion labs, implications for validating inertial magnetic sensors," *Gait & Posture*, vol. 29, no. 4, pp. 535–541, Jun. 2009, doi: 10.1016/j.gaitpost.2008.12.004.

[8] M. J. Tobin, "Basing Respiratory Management of COVID-19 on Physiological Principles," *Am. J. Respir. Crit. Care Med.*, vol. 201, no. 11, pp. 1319–1320, Apr. 2020, doi: 10.1164/rccm.202004-1076ED.

[9] M. H. Brooke *et al.*, "Duchenne muscular dystrophy: Patterns of clinical progression and effects of supportive therapy," *Neurology*, vol. 39, no. 4, pp. 475–481, Apr. 1989.

[10] M. Kohler, C. F. Clarenbach, L. Boni, T. Brack, E. W. Russi, and K. E. Bloch, "Quality of Life, Physical Disability, and Respiratory Impairment in Duchenne Muscular Dystrophy,"

Am. J. Respir. Crit. Care Med., vol. 172, no. 8, pp. 1032–1036, 2005, doi: 10.1164/rccm.200503-322OC.

[11] A. Houssein, D. Ge, S. Gastinger, R. Dumond, and J. Prioux, "Estimation of respiratory variables from thoracoabdominal breathing distance: a review of different techniques and calibration methods," *Physiol. Meas.*, vol. 40, no. 3, pp. 03TR01-03TR01, 2019, doi: 10.1088/1361-6579/ab0b63.

[12] R. Pierce, "Spirometry: an essential clinical measurement," *Aust. Fam. Physician*, vol. 34, no. 7, pp. 535–9, 2005.

[13] J Askanazi, P A Silverberg, R J Foster, A I Hyman, J Milic-Emili, and J M Kinney, "Effects of respiratory apparatus on breathing pattern," *J. Appl. Physiol.*, vol. 48, no. 4, pp. 577–580, 1980, doi: 10.1152/jappl.1980.48.4.577.

[14] P. D. Sly and C. F. Robertson, "A Review of Pulmonary Function Testing in Children," *J. Asthma*, vol. 27, no. 3, pp. 137–147, 1990, doi: 10.3109/02770909009073314.

[15] K. Konno and J. Mead, "Measurement of the separate volume changes of rib cage and abdomen during breathing," *J. Appl. Physiol.*, vol. 22, no. 3, pp. 407–422, 1967, doi: 10.1152/jappl.1967.22.3.407.

[16] F. D. McCool, J. Wang, and K. L. Ebi, "Tidal Volume and Respiratory Timing Derived From a Portable Ventilation Monitor," *Chest*, vol. 122, no. 2, pp. 684–691, 2002, doi: 10.1378/chest.122.2.684.

[17] S. Motamedi-Fakhr *et al.*, "Evaluation of the agreement of tidal breathing parameters measured simultaneously using pneumotachography and structured light plethysmography," *Physiol. Rep.*, vol. 5, no. 3, p. e13124, 2017, doi: 10.14814/phy2.13124.

[18] Q. Yuan, E. Asadi, Q. Lu, G. Yang, and I.-M. Chen, "Uncertainty-Based IMU Orientation Tracking Algorithm for Dynamic Motions," *IEEE/ASME Trans. Mechatronics*, vol. 24, no. 2, pp. 872–882, 2019, doi: 10.1109/TMECH.2019.2892069.

[19] L. Jin, H. Zhang, and C. Ye, "Camera Intrinsic Parameters Estimation by Visual-Inertial Odometry for a Mobile Phone With Application to Assisted Navigation," *IEEE/ASME Trans. Mechatronics*, vol. 25, no. 4, pp. 1803–1811, 2020, doi: 10.1109/TMECH.2020.2997606.

[20] F. Aghili and A. Salerno, "Driftless 3-D Attitude Determination and Positioning of Mobile Robots By Integration of IMU With Two RTK GPSs," *IEEE/ASME Trans. Mechatronics*, vol. 18, no. 1, pp. 21–31, 2013, doi: 10.1109/TMECH.2011.2161485.

[21] G. Lu and F. Zhang, “IMU-Based Attitude Estimation in the Presence of Narrow-Band Noise,” *IEEE/ASME Trans. Mechatronics*, vol. 24, no. 2, pp. 841–852, 2019, doi: 10.1109/TMECH.2019.2896130.

[22] T. Reinvuo, M. Hannula, H. Sorvoja, E. Alasaarela, and R. Myllyla, “Measurement of respiratory rate with high-resolution accelerometer and emfit pressure sensor,” in *Proc. 2006 IEEE Sensors Applications Symp.*, Houston, TX, USA, 2006, pp. 192–195. doi: 10.1109/SAS.2006.1634270.

[23] Ja-Woong Yoon, Yeon-Sik Noh, Yi-Suk Kwon, Won-Ki Kim, and Hyung-Ro Yoon, “Improvement of Dynamic Respiration Monitoring Through Sensor Fusion of Accelerometer and Gyro-sensor,” *J. Electr. Eng. Technol.*, vol. 9, no. 1, pp. 334–343, 2014, doi: 10.5370/JEET.2014.9.1.334.

[24] A. R. Fekr, K. Radecka, and Z. Zilic, “Tidal volume variability and respiration rate estimation using a wearable accelerometer sensor,” *Proc. 2014 International Conf. on Wireless Mobile Communication and Healthcare*, pp. 1–6, Nov. 2014.

[25] A. Siqueira, A. F. Spirandeli, R. Moraes, and V. Zarzoso, “Respiratory Waveform Estimation From Multiple Accelerometers: An Optimal Sensor Number and Placement Analysis,” *IEEE J. Biomed. Health Inform.*, vol. 23, no. 4, pp. 1507–1515, 2019, doi: 10.1109/JBHI.2018.2867727.

[26] G. Elshafie, P. Kumar, S. Motamedi-Fakhr, R. Iles, R. C. Wilson, and B. Naidu, “Measuring changes in chest wall motion after lung resection using structured light plethysmography: a feasibility study,” *Interact. J. Thorac. Cardiovasc. Surg.*, vol. 23, no. 4, pp. 544–547, 2016, doi: 10.1093/icvts/ivw185.

[27] K. Tukanova, E. Papi, S. Jamel, G. B. Hanna, A. H. McGregor, and S. R. Markar, “Assessment of chest wall movement following thoracotomy: a systematic review,” *J. Thorac. Dis.*, vol. 12, no. 3, pp. 1031–1040, 2020, doi: 10.21037/jtd.2019.12.93.

[28] G. Ligorio and A. M. Sabatini, “A Novel Kalman Filter for Human Motion Tracking With an Inertial-Based Dynamic Inclinometer,” *IEEE Trans. Biomed. Eng.*, vol. 62, no. 8, pp. 2033–2043, 2015, doi: 10.1109/TBME.2015.2411431.

[29] F. L. Markley, “Lessons Learned,” *J. Astronaut. Sci.*, vol. 57, no. 1, pp. 3–29, 2009, doi: 10.1007/BF03321491.

[30] D. Simon, *Optimal state estimation*. Hoboken, N.J.: Wiley-Interscience, 2006.

[31] J. Sedlak, “Comparison of Kalman filter and optimal smoother estimates of spacecraft attitude,” presented at the Flight Mechanics/Estimation Theory Symposium, Greenbelt, MD., 1994.

[32] P. I. Corke, *Robotics, vision and control*, Second, Completely revised, Extended and Updated edition.. Cham, Switzerland: Springer, 2017.

[33] E. A. Hooker, D. J. O’Brien, D. F. Danzl, J. A. Barefoot, and J. E. Brown, “Respiratory rates in emergency department patients,” *J. Emerg. Med.*, vol. 7, no. 2, pp. 129–132, 1989, doi: 10.1016/0736-4679(89)90257-6.

[34] S. Fleming *et al.*, “Normal ranges of heart rate and respiratory rate in children from birth to 18 years of age: a systematic review of observational studies,” *Lancet*, vol. 377, no. 9770, pp. 1011–1018, 2011, doi: 10.1016/S0140-6736(10)62226-X.

[35] R. Panos, *A COPD Primer*. Warsaw ; Berlin: De Gruyter Open Poland, 2016.

[36] J. S. Whittle, I. Pavlov, A. D. Sacchetti, C. Atwood, and M. S. Rosenberg, “Respiratory support for adult patients with COVID-19,” *JACEP Open*, vol. 1, no. 2, pp. 95–101, 2020, doi: 10.1002/emp2.12071.

[37] G. Johnson, P. Pianosi, and R. Rajamani, “Estimation of three dimensional thoracoabdominal displacements during respiration using intertial measurement units,” *IEEE/ASME Trans. Mechatronics*, vol. Accepted for publication.

[38] G. Johnson, Y. Wang, and R. Rajamani, “Real-Time Detection of Food Consumption Activities Using Wearable Wireless Sensors,” *Proc. 2019 American Control Conf.*, pp. 3450–3455, Jul. 2019.

[39] J. Sedlak, “Comparison of Kalman filter and optimal smoother estimates of spacecraft attitude,” *Proc. Flight Mechanics/Estimation Theory Symp.*, 1994.

[40] T. Brack, A. Jubran, and M. J. Tobin, “Effect of resistive loading on variational activity of breathing,” *Am. J. Respir. Crit. Care Med.*, vol. 157, no. 6, pp. 1756–1763, 1998, doi: 10.1164/ajrccm.157.6.9704114.

[41] J. Hammer and C. J. L. Newth, “Assessment of thoraco-abdominal asynchrony,” *Paediatr. Respir. Rev.*, vol. 10, no. 2, pp. 75–80, 2009, doi: 10.1016/j.prrv.2009.02.004.

[42] M. J. Tobin, T. S. Chadha, G. Jenouri, S. J. Birch, H. B. Gazeroglu, and M. A. Sackner, “Breathing Patterns: 2. Diseased Subjects,” *Chest*, vol. 84, no. 3, pp. 286–294, Sep. 1983, doi: 10.1378/chest.84.3.286.

[43] G. K. Prisk, J. Hammer, and C. J. Newth, “Techniques for measurement of thoracoabdominal asynchrony,” *Pediatr. Pulm.*, vol. 34, no. 6, pp. 462–472, 2002, doi: 10.1002/ppul.10204.

[44] J. Hammer, C. J. L. Newth, and T. W. Deakers, “Validation of the phase angle technique as an objective measure of upper airway obstruction,” *Pediatr. Pulm.*, vol. 19, no. 3, pp. 167–173, 1995, doi: 10.1002/ppul.1950190305.

[45] M A Sackner *et al.*, “Calibration of respiratory inductive plethysmograph during natural breathing,” *J. Appl. Physiol.*, vol. 66, no. 1, pp. 410–420, 1989, doi: 10.1152/jappl.1989.66.1.410.

[46] T. Rahman, R. Page, C. Page, J.-R. Bonnefoy, T. Cox, and T. H. Shaffer, “pneuRIPTM: A Novel Respiratory Inductance Plethysmography Monitor,” *J. Med. Devices*, vol. 11, no. 1, pp. 0110101–0110106, 2017, doi: 10.1115/1.4035546.

[47] F. Laghi and M. J. Tobin, “Chapter 4. Indications for Mechanical Ventilation,” in *Principles and Practice of Mechanical Ventilation*, 3rd ed., M. J. Tobin, Ed. New York, NY: The McGraw-Hill Companies, 2013.

[48] N. R. MacIntyre, “The Ventilator Discontinuation Process: An Expanding Evidence BaseDiscussion,” *Respir. Care*, vol. 58, no. 6, pp. 1074–1086, Jun. 2013, doi: 10.4187/respcare.02284.

[49] M. J. Tobin, F. Laghi, and A. Jubran, “Ventilatory failure, ventilator support, and ventilator weaning,” *Compr. Physiol.*, vol. 2, no. 4, pp. 2871–2921, 2012.

[50] M. J. Tobin, W. Perez, S. M. Guenther, R. F. Lodato, and D. R. Dantzker, “Does rib cage-abdominal paradox signify respiratory muscle fatigue?,” *Journal of Applied Physiology*, vol. 63, no. 2, pp. 851–860, Aug. 1987, doi: 10.1152/jappl.1987.63.2.851.

[51] W. Willett, *Nutritional epidemiology*, Third edition.. Oxford ; New York: Oxford University Press, 2013.

[52] V. Kipnis *et al.*, “Structure of Dietary Measurement Error: Results of the OPEN Biomarker Study,” *American Journal of Epidemiology*, vol. 158, no. 1, pp. 14–21, Jul. 2003, doi: 10.1093/aje/kwg091.

[53] A. J. Moshfegh *et al.*, “The US Department of Agriculture Automated Multiple-Pass Method reduces bias in the collection of energy intakes,” *The American Journal of Clinical Nutrition*, vol. 88, no. 2, pp. 324–332, Aug. 2008, doi: 10.1093/ajcn/88.2.324.

[54] A. Schatzkin *et al.*, “A comparison of a food frequency questionnaire with a 24-hour recall for use in an epidemiological cohort study: results from the biomarker-based Observing

Protein and Energy Nutrition (OPEN) study," *International Journal of Epidemiology*, vol. 32, no. 6, pp. 1054–1062, Dec. 2003, doi: 10.1093/ije/dyg264.

[55] D. Lansky and K. D. Brownell, "Estimates of food quantity and calories: errors in self-report among obese patients," *The American Journal of Clinical Nutrition*, vol. 35, no. 4, pp. 727–732, Apr. 1982, doi: 10.1093/ajcn/35.4.727.

[56] R. K. Johnson, R. P. Soultanakis, and D. E. Matthews, "Literacy and Body Fatness are Associated with Underreporting of Energy Intake in US Low-Income Women Using the Multiple-Pass 24-hour Recall: A Doubly Labeled Water Study," *Journal of the American Dietetic Association*, vol. 98, no. 10, pp. 1136–1140, Oct. 1998, doi: 10.1016/S0002-8223(98)00263-6.

[57] J. R. Hebert *et al.*, "Gender Differences in Social Desirability and Social Approval Bias in Dietary Self-report," *American Journal of Epidemiology*, vol. 146, no. 12, pp. 1046–1055, Dec. 1997, doi: 10.1093/oxfordjournals.aje.a009233.

[58] V. Kipnis *et al.*, "Bias in dietary-report instruments and its implications for nutritional epidemiology," *Public Health Nutrition*, vol. 5, no. 6a, pp. 915–923, Dec. 2002, doi: 10.1079/PHN2002383.

[59] L. J. Harnack and M. A. Pereira, "Strategies for Improving the Validity of the 24-hour Dietary Recall and Food Record Methods," in *Advances in the Assessment of Dietary Intake*, CRC Press, 2017.

[60] J. L. Crassidis, F. L. Markley, and Y. Cheng, "Survey of Nonlinear Attitude Estimation Methods," *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 1, pp. 12–28, Jan. 2007, doi: 10.2514/1.22452.

[61] Y. Wang and R. Rajamani, "Direction cosine matrix estimation with an inertial measurement unit," *Mechanical Systems and Signal Processing*, vol. 109, pp. 268–284, Sep. 2018, doi: 10.1016/j.ymssp.2018.02.038.

[62] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.

[63] B. Wie, *Space vehicle dynamics and control*, 2nd ed.. Reston, Va., Reston, VA: American Institute of Aeronautics and Astronautics, 2008.

[64] D. Choukroun, "Novel methods for attitude determination using vector observations," Technion - Israel Institute of Technology, 2003.

[65] H. E. Knoepfel, “Magnetic Fields: A Comprehensive Theoretical Treatise for Practical Use,” *American Journal of Physics*, vol. 69, no. 4, pp. 525–525, 2001, doi: 10.1119/1.1351150.

[66] R. Madson, “Position Estimation Using Magnetic Fields,” Ph.D., University of Minnesota, United States -- Minnesota, 2018. Accessed: Jun. 15, 2021. [Online]. Available: http://www.proquest.com/dissertations/docview/2334392987/abstract/1D0823AC04B14B89PQ/7

[67] K. P. Bennett and C. Campbell, “Support vector machines: hype or hallelujah?,” *SIGKDD Explor. Newsl.*, vol. 2, no. 2, pp. 1–13, Dec. 2000, doi: 10.1145/380995.380999.

[68] M. Kuhn and K. Johnson, *Applied predictive modeling*. New York: Springer, 2013.

[69] S. Albawi, T. A. Mohammed, and S. Al-Zawi, “Understanding of a convolutional neural network,” in *2017 International Conference on Engineering and Technology (ICET)*, Aug. 2017, pp. 1–6. doi: 10.1109/ICEngTechnol.2017.8308186.

[70] R. Sanchez-Iborra and A. F. Skarmeta, “TinyML-Enabled Frugal Smart Objects: Challenges and Opportunities,” *IEEE Circuits and Systems Magazine*, vol. 20, no. 3, pp. 4–18, thirdquarter 2020, doi: 10.1109/MCAS.2020.3005467.

[71] R. David *et al.*, “TensorFlow Lite Micro: Embedded Machine Learning on TinyML Systems,” *arXiv:2010.08678 [cs]*, Mar. 2021, Accessed: Mar. 30, 2021. [Online]. Available: http://arxiv.org/abs/2010.08678

[72] J. Dai, “Real-time and accurate object detection on edge device with TensorFlow Lite,” *J. Phys.: Conf. Ser.*, vol. 1651, p. 012114, Nov. 2020, doi: 10.1088/1742-6596/1651/1/012114.

[73] J. V. M. Sousa, V. R. de Almeida, A. A. Saraiva, D. B. S. Santos, P. M. C. Pimentel, and L. L. de Sousa, “Classification of Pneumonia images on mobile devices with Quantized Neural Network,” *RSD*, vol. 9, no. 10, Art. no. 10, Sep. 2020, doi: 10.33448/rsd-v9i10.8382.

[74] E. W. Weisstein, “Harmonic Addition Theorem.” https://mathworld.wolfram.com/HarmonicAdditionTheorem.html (accessed Jun. 08, 2021).

[75] A. Krizhevsky and G. Hinton, “Learning multiple layers of features from tiny images.,” 2009.

[76] W. Wang, Y. Yang, X. Wang, W. Wang, and J. Li, “Development of convolutional neural network and its application in image classification: a survey,” *OE*, vol. 58, no. 4, p. 040901, Apr. 2019, doi: 10.1117/1.OE.58.4.040901.

[77] R. C. Çalik and M. F. Demirci, “Cifar-10 Image Classification with Convolutional Neural Networks for Embedded Systems,” in *2018 IEEE/ACS 15th International Conference on Computer Systems and Applications (AICCSA)*, Oct. 2018, pp. 1–2. doi: 10.1109/AICCSA.2018.8612873.

[78] P. Foret, A. Kleiner, H. Mobahi, and B. Neyshabur, “Sharpness-Aware Minimization for Efficiently Improving Generalization,” Apr. 2021. Accessed: Jun. 27, 2021. [Online]. Available: http://arxiv.org/abs/2010.01412

[79] National Safety Council and National Safety Council Research and Statistics Department, *Injury facts*. Itasca, IL: National Safety Council, 2015.

[80] E. O. Olsen, R. A. Shults, and D. K. Eaton, “Texting While Driving and Other Risky Motor Vehicle Behaviors Among US High School Students,” *Pediatrics*, vol. 131, no. 6, pp. e1708–e1715, Jun. 2013, doi: 10.1542/peds.2012-3462.

[81] Y. Wang, J. Yang, H. Liu, Y. Chen, M. Gruteser, and R. P. Martin, “Sensing vehicle dynamics for determining driver phone use,” in *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*, New York, NY, USA, Jun. 2013, pp. 41–54. doi: 10.1145/2462456.2464447.

[82] Z. He, J. Cao, X. Liu, and S. Tang, “Who Sits Where? Infrastructure-Free In-Vehicle Cooperative Positioning via Smartphones,” *Sensors (Basel)*, vol. 14, no. 7, pp. 11605–11628, Jun. 2014, doi: 10.3390/s140711605.

[83] J. Yang *et al.*, “Sensing Driver Phone Use with Acoustic Ranging through Car Speakers,” *IEEE Transactions on Mobile Computing*, vol. 11, no. 9, pp. 1426–1440, Sep. 2012, doi: 10.1109/TMC.2012.92.

[84] C. Bo *et al.*, “Detecting Driver’s Smartphone Usage via Nonintrusively Sensing Driving Dynamics,” *IEEE Internet of Things Journal*, vol. 4, no. 2, pp. 340–350, Apr. 2017, doi: 10.1109/JIOT.2016.2552399.

[85] H. Chu, V. Raman, J. Shen, A. Kansal, V. Bahl, and R. R. Choudhury, “I am a smartphone and I know my user is driving,” in *2014 Sixth International Conference on Communication Systems and Networks (COMSNETS)*, Jan. 2014, pp. 1–8. doi: 10.1109/COMSNETS.2014.6734870.

[86] J. G. Elias, “Driver handheld computing device lock-out,” US8706143B1, Apr. 22, 2014 Accessed: Jun. 04, 2021. [Online]. Available: https://patents.google.com/patent/US8706143B1/en

[87] R. Rajamani, *Vehicle dynamics and control*, 2nd ed.. New York: Springer, 2012.

[88] T. Shim and C. Ghike, “Understanding the limitations of different vehicle models for roll dynamics studies,” *Vehicle System Dynamics*, vol. 45, no. 3, pp. 191–216, Mar. 2007, doi: 10.1080/00423110600882449.

[89] V. T. Vu, O. Sename, L. Dugard, and P. Gaspar, “Enhancing roll stability of heavy vehicle by LQR active anti-roll bar control using electronic servo-valve hydraulic actuators,” *Vehicle System Dynamics*, vol. 55, no. 9, pp. 1405–1429, Sep. 2017, doi: 10.1080/00423114.2017.1317822.

[90] C. E. Beal and C. Boyd, “Coupled lateral-longitudinal vehicle dynamics and control design with three-dimensional state portraits,” *Vehicle System Dynamics*, vol. 57, no. 2, pp. 286–313, Feb. 2019, doi: 10.1080/00423114.2018.1467019.

[91] V. K. Ingle, *Digital signal processing using MATLAB*, 2nd ed.. Toronto, Ontario, Canada: Nelson, 2007.