# On Quaternions and Activity Classification Across Sensor Domains

Jacob H. Dennis

Thesis submitted to the Faculty of the
Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements for the degree of

Master of Science
in
Computer Engineering

Thomas L. Martin, Chair
Mark T. Jones, Co-Chair
Nicholas F. Polys

December 5, 2014
Blacksburg, Virginia

Keywords: Activity Classification, Wearable Computing, Body-Model

# On Quaternions and Activity Classification Across Sensor Domains

Jacob H. Dennis

(ABSTRACT)

Activity classification based on sensor data is a challenging task. Many studies have focused on two main methods to enable activity classification; namely sensor level classification and body-model level classification. This study aims to enable activity classification across sensor domains by considering an e-textile garment and provide the groundwork for transferring the e-textile garment to a vision-based classifier. The framework is comprised of three main components that enable the successful transfer of the body-worn system to the vision-based classifier. The inter-class confusion of the activity space is quantified to allow an ideal prediction of known class accuracy for varying levels of error within the system. Methods for quantifying sensor and garment level error are undertaken to identify challenges specific to a body-worn system. These methods are then used to inform decisions related to the classification accuracy and threshold of the classifier. Using activities from a vision-based system known to the classifier, a user study was conducted to generate an observed set of activities from the body-worn system. The results indicate that the vision-based classifier used is user-independent and can successfully handle classification across sensor domains.

# Acknowledgments

I am indebted to many people for the completion of this work:

To the participants of the user study who endured repetitive activities to collect the data set.

To the many friends in the CCM lab for helping along the way.

To Matt, my intrepid reviewer and best friend.

Finally, to Dr. Tom Martin for his role as advisor throughout graduate school and the guidance he has provided during that time. Dr. Mark Jones and Dr. Nicholas Polys for taking the time to serve on my committee and help guide me though the research process.

This material is based in part upon work supported by the National Science Foundation under Grant Number IIS-1116669. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Motivation

Wearable computing touches on many aspects of everyday life - self-quantification through fitness monitoring devices, distributed cognition through interaction with smart watches, and integration with mobile devices. As the field of e-textiles progresses, new sensor systems and styles become available. When changes occur to the sensors or placement of a system, it is desirable to limit the amount of re-training and data collection necessary for an activity classification system. This study explores methods that allow effective classification across sensor domains to enable ambulatory monitoring. Specifically, this study targets loose-fitting body-worn systems to encourage greater compliance of users tracking physical therapy routines or at home health monitoring with automatic activity logging. The work described within builds on Blake (2014) using an analysis of sensor error, activity overlap, and filter discussions to understand the challenges of moving toward a body-worn system from a vision-based classifier.

## 1.2 Methodology

The goal of this paper is to show that the classification system investigated is sensor-independent. Blake (2014) details the specifics of a body-model and classifier used as the basis for this thesis. Using dynamic time warping and an alphabet matching scheme Navarro (2001), an observed activity can be matched to a known set of activities in a manner that is both user-independent and robust to user variability. The classifier was shown to be effective at labeling unknown activities as unknown within a vision-based domain.

This study has been built on the model-based approach in Blake (2014) by utilizing an inertial measurement unit (IMU)-based loose-fitting jumpsuit to collect a 20 activity data-set; the mix of activities ranged from household to fitness oriented. The collected dataset was compared to two vision-based systems, Kinect and Qualysis Motion Capture.

Banos et al. (2012), Forster et al. (2009), Ogris et al. (2008), and others have provided methods to transition between sensor domains. However, where these studies applied transformation operations from one sensor domain to the next, this paper constructs the necessary framework to investigate the transition to the body-model in Blake (2014).

The high-level approach undertaken to investigate sensor-independence of the classifier is structured into two focus areas; sensor domain, and classifier. This split enables the discussion of sensor specific challenges, which may be disjoint from the classifier. However, the sensor level discussion informs modifications at the classifier level, which aims to provide insight into areas where the classifier experiences classification challenges due to the underlying sensor system.

***Sensor Domain***

The exploration began with the construction of a synthetic activity to provide an intuitive grasp on how the classifier in Blake (2014) operates. Using this activity the allowable extremes of variation could then be systematically set and investigated. This led to the development of noise estimates, which then informed the designer on the accuracy of various body-worn systems.

Building on the synthetic activity, the underlying measurement error and noise contributions of the body-worn system were explored. Filters were designed and tested given simplifying assumptions on the velocity of the movements within the data-set. The testing was completed using static and dynamic error calculations. Examination of measurement error was carried out at both the sensor level domain and the quaternion level domain. The static testing was carried out with a PVC assembly to maintain stable positioning of the sensors during a test, while the dynamic tests were carried out on the synthetic activity.

***Classifier Domain***

During the measurement and error classification investigation, it was realized that the body-worn system is unable to provide the necessary fidelity measurement to facilitate accurate classification. Therefore, as a requirement in transitioning to a vision-based classifier, an error function was created that incorporates noise and measurement error of a body-worn system. This error function provides the intermediary step between the sensors and the classifier, allowing the classifier to remain unchanged while the sensor systems are exchanged. A side effect of this error function augmentation is a reduced accuracy of the classifier,leading

to the development of an activity intersection algorithm to predict expected accuracy.

This study utilizes a combination of the above methods, as well as a system comparison with respect to the fit of the garment by examining the overall accuracy between two different versions of the body-worn system.

## 1.3 Contributions

This work presents an analysis of a vision-based system with an emphasis on crossing the sensor domain to a wearable system. It branches out from prior works that operate in a single sensor domain and instead utilizes the methods of Blake (2014) to bridge the gap from a vision-based to a body-worn system. The sensor-independence is shown by utilizing a vision-based classifier and matching a set of body-worn activities to that classifier without re-training for the specific sensor domain. By showing the sensor-independence of the Model-based User-independent Activity Classifier (MUAC), the framework presented will allow for novel e-textile inspired sensors to be incorporated for everyday activity recognition in the future.

## 1.4 Paper Organization

The following chapters will outline the particular steps required to transition the vision-based classifier to one in a body-worn environment. Chapter 2 provides a brief background on the previous works in the area of activity classification, as well as, an overview of quaternions and outlines the assumptions used for the body-model and the specifics regarding the body-worn system. Chapter 3 presents the methodology and experimental analysis of the current system illustrated using a simple test activity. Chapter 4 describes the construction of the current prototype and changes from previous versions, and the user study conducted to validate the system. Chapter 5 presents the results of the user study. Finally, Chapter 6 concludes the work.

# Chapter 2

# Background

Designing a wearable ubiquitous activity recognition system involves an understanding of how the sensors interact with the wearer (Gemperle et al.), the perception of the added intrusion to the wearability of the system, and effects of sensor system changes. This chapter reviews the motivation for bridging sensor systems and will provide an overview of the prior studies that have laid the groundwork for such a system.

## 2.1 Wearability

Wearable computing adds layers of consideration to human computer interaction models (HCI). Physical, cognitive, and psychological variables must be considered within a large design space; appropriately considering these additional facets of is an evolving process. Social acceptance, comfort, user interaction and expectation are primary considerations.

### 2.1.1 Style

As with any article of clothing, style plays an important role in deciding what is worn on a day to day basis. Sensor placement is another factor critical to the wearable nature of a system. Stylistically complex and functionally simple activity monitoring systems exist; however, these are geared toward a finite market in which there is already a plethora of such devices. A large component of these devices are focused on health-conscious and fitness-minded individuals. Notable brands of fitness trackers include the FitBit, Jawbone, and Nike Fuse, which make up approximately 97 % of the current market. These products are quite simple in functionality; they measure step count, heart-rate, and other physiological markers to provide a fitness

metric for the general populus.

However, these products fall short of providing the information that could be added if a few additional sensors were to be strategically placed or embedded within clothing. How are these sensors worn or placed, and what would the resulting outfit look like?

Other systems that are geared toward high-fidelity movement include the Xsens MVN suit, and motion capture vision-based systems. These systems bypass fit related issues and use cat-suit like garments with the sensors affixed to the suit, or Velcro strapping used to hold sensors in place on the subject of interest. While these systems provide high-fidelity movement information, they are generally limited to laboratory environments or controlled settings, with the exception of the XSENS.

To provide a system that is truly wearable, the research team focused on a loose fitting garment for the tests and experiments considered here. The garments have three sizes to mitigate the bunching effects of a "one-size-fit-all" garment. With a system that is designed with a loose fit in mind, new sensors and systems can more easily transition to body worn garments as well as provide the desired functionality while maintaining user compliance. While the current prototype lacks a stylistic design that would enable users to wear the system without some level of embarrassment, the overarching goal to maintain the loose nature of the garment was successful.

### 2.1.2 Garment Fit

In addition to style, the fit of a garment must be considered in wearable systems. Ideally, the system should move with clothing when the system is embedded within or attached to the clothing; consequently, using one-size-fits-all garments introduces unnecessary error influences that must be overcome. The system used for this study is sized appropriately as "small", "medium" and "large" to help avoid these negative effects. A previous version of the system was made to fit a large portion of the population by using adjustment points. However, these adjustment points proved to be a contributor of the system noise. The accuracy of the sensor to track the limb movement for all possible positions is of special importance when considering sensors embedded within a garment.

As garment fit plays a role in exterior mounted sensors, purpose built shirts, or wrist worn devices, researchers are experimenting with fabric-inspired sensors. To this end, Gioberto and Dunne (2012) have investigated fabric strain sensors composed of conductive fibers woven into an interlock stitch, which allow movement of the fabric backbone without compromising the drape or feel.

## 2.2 Sensor Domain Mapping

Several groups have published techniques describing how to map effectively from one sensor domain to another, albeit at a cost to overall accuracy.(Banos et al., 2012),(Zinnen et al., 2009a),(Blanke and Schiele, 2010). These methods provide the ability to transition from one sensor domain to another for validation and experimentation of new systems. While this ability is useful, the collected data is still maintained in various sensor domains without a unifying mapping.

### 2.2.1 Sensor-Based Designs

Sensor-based systems have two broad categories of operation: single domain operation where activities are mapped to single sensor input curves, and heterogeneous systems where various sensors are combined. Integration issues, base-line shift, and placement irregularities must be overcome, depending on the sensor type. Learning methods are then employed to arrive at suitable feature vectors for the activity space under observation. Sensor placement independence can be achieved by using datasets for the same activity with varying sensor positions. Lester et al. (2006) illustrates the sensor placement independence by classifying an 8 activity set. The generic classifier in the paper achieves appreciable results when using only a subset of the sensors, thereby allowing a user to wear(or not wear) sensors that they may find intrusive in certain social or environmental contexts. While this result is interesting, the overall activity space is limited. Additionally, the confusion between disjoint activities (sitting/standing) may limit the applicability of this approach. Other approaches to sensor shift due to day-to-day wear involve unsupervised calibration to account for the sensor shift, as shown by Forster et al. (2009). The calibration is an event driven scheme that becomes active once classification accuracy drops below a known threshold. With the advent of various sensors within cellphones, watches, and shoes; it becomes necessary to provide translation to known classifier domains. Banos et al. (2012) provides a MIMO mapping from a known sensor domain to that of a new domain. This is performed by learning the original activities in one domain and then providing the new domain activities as the end mapping, providing the necessary conditions to create a transform matrix and successfully map the activity classifier to the new domain. However, this requires a set of know "good" activities in the other domain, which may not be available. The transform matrix is a high-order expansion mapping that requires re-training for each additional sensor mapping.

These sensor based designs focus on varying challenges of body-worn systems. Robustness to sensor placement, calibration, and sensor domain translation are all necessary aspects of a successful technology.

However, these methods do not guarantee user-independence. Therefore, this study examines the benefits of the body-model designs.

### 2.2.2 Body Model Designs

Body-model defined activity sequences are beneficial with respect to sensor based features as they are generally more robust to sensor displacement. The body-model is normally defined in a linked kinematic form, which also provides an abstraction layer allowing for more complex dynamic body models to be defined. Zinnen et al. (2009a) compares signal-oriented and model-based approaches; the authors show that using a body-model based approach leads to improved results. Zinnen et al. (2009b) present body-model derived primitives to train a classifier in a sequence of car maintenance tasks. The paper uses motion and location primitives to develop the necessary feature vectors for classification. The ability to define a "task" as sequences of the relationship between limbs or joint angles is the most similar to Blake (2014); however, the authors in Zinnen et al. (2009b) use an abstraction class of motion primitives which form higher level concepts of movement (push/pull, twist, direction) of the limbs respective to a reference point. These higher level primitives are then combined with relative location and velocity features around a vehicle to provide a feature vector of 373 elements. The feature vector is then used to determine the activity within a car maintenance activity set of 20 classes. The higher level primitives are divergent from the focus of Blake (2014) which seeks to provide an abstraction layer that provides sensor-independence as well as user independence. This paper seeks to provide the necessary framework to bridge this gap and illustrate the sensor-independence with the IMU jumpsuit (Figure 2.1).

With this in mind, efforts were focused on providing a body-model based framework that provides for inherent sensor independence, building on the classification from Blake (2014). The benefits of the model based design are two-fold: intra-class variability in body structure is minimized, and limb importance per activity can be mapped to help provide robustness in intra-class variability. Additionally, sensor-independence can be gained, provided that a given a sensor domain can be mapped to the quaternion space. This is fairly straightforward for IMU and vision based systems. With a strain based system, angular displacement of the joints is necessary. The angular displacement of the joints can be generated by defining a mapping from a resistance or strain measurement to an angular displacement. The following sections we will delve into the specifics of the quaternion system as it applies to a loose-fitting body worn system in Section 2.3. Using the methods and analysis within Chapter 3, a first-order approximation of the threshold adjustment to account for increase in error of a body-worn system, is provided. Section 3.3.1 presents a method to analyze

Figure 2.1: Loose-Fitting Jumpsuit V2

inter-class confusion and the trade-off in accuracy of a body-worn system. This analysis is then discussed with results in Chapter 5.

## 2.3 Quaternion Application

This section provides a brief introduction of quaternions and the the underlying math involved. The chapter presents the quaternion mapping formulas and explains the body-model based construction of pose sequences as it applies to the jumpsuit used in this study.

## 2.4 Basics

Mathematical body rotations in space can be carried out in various ways, the most familiar method being the fairly intuitive Euler angle rotation matrices. Using this method, body rotations are calculated given an axis order and angles of rotation about those axes. These concepts are vitally fundamental to this study.

Before delving into the underlying quaternion methods, it is important to understand the essential design choices. When defining the rotation of an object, it is necessary to define the mode of rotation and which axis sequence the object is rotated about. By defining a rotation order, a unique rotation in space can be determined.

The mode that frames system reference can be intrinsic or extrinsic. The rotation mode used within the Model-based User-independent Activity Classifier (MUAC) and the body-worn system was based on a locally defined body axis (*instrinsic*). Using this method, local rotations can be mapped back to the original global axis. This allows a way to reference the rotations with respect to a central point, in the form of a linked body. The selected order of operations for the rotations is ZYX; or yaw ($\phi$) , pitch ($\theta$) , and roll ($\psi$) as is referred to in this literature. Equation 2.1 presents the transformation from the intrinsic Euler rotation of the locally defined axis to the quaternion form:

$$\begin{aligned} w &= \cos\phi/2 * \cos\theta/2 * \cos\psi/2 + \sin\phi/2 * \sin\theta/2 * \sin\psi/2 \\ x &= -\sin\phi/2 * \sin\theta/2 * \cos\psi/2 + \cos\phi/2 * \cos\theta/2 * \sin\psi/2 \\ y &= \cos\phi/2 * \sin\theta/2 * \cos\psi/2 + \sin\phi/2 * \cos\theta/2) * \sin\psi/2 \\ z &= -\cos\phi/2 * \sin\theta/2 * \sin\psi/2 + \sin\phi/2 * \cos\theta/2 * \cos\psi/2 \end{aligned} \tag{2.1}$$

Additionally, for each quaternion (Q) calculated, Q is put in reference to a particular segment which is *different* than using Equation 2.1. Example 1 illustrates the difference of the body-model quaternion and a generic quaternion, as well as the method by which the linked segments in the body-model are computed with respect to the parent.

As this study compares results across sensor domains, which will be described in Chapter 5, it is necessary to understand how the mapping from different domains to the nine-dimensioned body-model is accomplished. Given a set of joint angles from the Kinect system the limb segments are defined as rotated vectors in space in quaternion form. For example, each limb segment rotation is turned into a series of rotations from the torso. The upper arm segment depends on one rotation relative to the torso, and the lower arm is a rotation

**Example 1** Example quaternion rotation

▷ Given two linked segments $S_1$ and $S_2$, the intrinsic rotation is defined as $\phi_1, \theta_1, \psi_1$ and $\phi_2, \theta_2, \psi_2$, respectively. First the quaternion rotation of $S_1$ and $S_2$ is computed.

▷ Steps:

1: $\phi_1 = \pi/10$ ; $\phi_2 = \pi/10$
2: $\theta_1 = \pi/15$ ; $\theta_2 = \pi/15$
3: $\psi_1 = \pi/5$ ; $\psi_2 = \pi/5$
4: $Q =< w, x, y, z >$
5: $S_1 =< 0.9393, 0.2880, 0.1463, 0.1161 >$ ▷ Resultant quaternion definition based on 2.1
6: $S_2 =< 0.9393, 0.2880, 0.1463, 0.1161 >$ ▷ Resultant quaternion definition based on 2.1

▷ Next, define the quaternion rotation of $S_2$ as the additional rotation from $S_1$. To accomplish this transformation, the quaternion conjugate and multiplication is used to compute the additional rotation of $S_2$ with respect to $S_1$.

7: $S_2 = \overline{S_1} * S_2$
8: $S_2 =< 1.0, 0.0, 0.0, 0.0 >$

relative to the upper arm segment. These rotations then form the body-model mapping, as described in (Blake, 2014). In a similar manner the motion capture system joint angles are computed based on the markers in 3-D space. The computed joint angles are then used in a similar manner to map to the nine-dimensioned body-model. It should be noted that the body-model mapping only considers the joint angles of the shoulders, hips, elbows and knees. However, the model is expandable as noted in Section 6.2. The Jumpsuit mapping is different as the sensors are affixed to the clothing and detect the rotation of the limbs, which is explained in further detail in Section 2.4.1.

### 2.4.1 Jumpsuit Specifics

The Jumpsuit is a body-worn loose-fitting garment. The inertial measurement units (IMU)s attached to the suit provide 3-D location information for the nine segments of the body-model. There are two necessary modifications of the quaternion information as provided from Equation 2.1:

1. Mapping the quaternions to be in reference to the torso quaternion

2. Reduction to 2-degrees of freedom on the limb segments.

To map the limb segments into the correct frame of reference, the heading of the torso is removed from both the torso and the upper limb segments in the body-model. The mapping aligns the torso along the y-axis, and the limb segments are then aligned to an offset from the reference quaternion. The mapping of

the system is illustrated in the psuedo-code shown in Example 2. With this correction, the local quaternions are mapped to a known state with the environment rotation removed.

**Example 2** Mapping the quaternions to be aligned to the frame of reference

1: $QC \leftarrow QZROT.Conj() * QuatNode$
▷ * Represents the quaternion product

After remapping the quaternions to remove the environmental rotation, the limb segment quaternions are reduced to 2-degrees of freedom by removing the roll component of the rotation. This removal is accomplished by computing the quaternion rotation from a Euclidean vector $v_1$ to $v_2$. The algorithm is explained in Blake (2014). Finally, because the system is a linked kinematic model, the absolute rotation of a limb is given as the difference between the parent segment and the limb. Using this approach, the quaternion representation for a limb segment is an additional rotation from the parent (Example 1).

# Chapter 3

# Analysis of Error: Sensors, Quaternions, and System

This chapter details the key components necessary to transition from a vision based classification system to a body-worn system. The differences between systems stem from wearability and accuracy. In body-worn systems the effects of clothing, such as those due to the fit of the garment and sensor artifacts, degrade the accuracy of the information being collected.

In order to examine these effects in a controlled manner, activities are built as a known body-model sequence. Constructing an activity from a known position and adding perturbations to the body model allows for modeling of measurement error and noise propagation withing the system and allows these issues to be corrected. To this end, this section details the key components necessary to transition from a vision based classification system to a body worn system. The methods described detail the characterization of the underlying sensor systems, the body-model, methods to reduce the error contributions at both the sensor and body-level, and the overlap of the activity space.

## 3.1 Simple Activity Creation Right Arm Sweep

Section 3.1 outlines the synthetic activity creation that allows investigations to evaluate the noise contributions of the filters and sensor measurement that follow. Section 3.2 evaluates the sensor domain error and its impact on the quaternion representation. Section 3.3.1 examines the overlap within the activity space which enables an understanding of the confusion between activities. Finally, Section 3.3.2 derives an

augmented error metric on the activity sequence to account for a noisier system. Taken as a whole, these sections describe the necessary considerations and evaluation framework to transition MUAC to a body-worn environment.

Error accumulation is examined as a function of segment error computed from a body worn frame. Where Lewis (2011) examined the error in the jumpsuit in comparison to an optical system, this study begins by examining the error accumulation inherent to the body-model definition itself. As Lewis (2011) illustrates, error accumulates in different limb segments. However, the experiments do not account for initial offset of the limb sensors. Therefore, this study examines error contribution for both a static pose and a synthetic activity. This synthetic activity was generated to examine the noise and error propagation of the body-model. Because the activity is built without error, the best fit quaternion can be determined. With this, the limits of the error can be generated, and thus the weights on the limbs determined. The activity chosen for this exercise involves the subject standing with the right arm out in front of the chest at a 90° angle parallel to the ground as illustrated in Figure 3.1. The sequence involves a rotation, keeping the arm straight, from the front to the side of the torso while maintaining the arm parallel to the floor (Figure 3.1). The arm sweep activity has been chosen because of its mathematically simple construction and its ability to examine error propagation combining the upper arm, forearm, and the torso.

Figure 3.2 illustrates the noise propagation from the torso to the limb segments. The experiment was chosen to show that, even with no noise in the synthetic activity on the limb segments there is still inherent noise from the parent. Given a bounded noise applied only to the torso segment, the error propagates from the construction of the body-model in a body-worn system. The error transfer from the parent limb to the child limb is important to understand, as it will accumulate in different manners for different sensor systems. A concrete example of how the accumulation changes across sensor domains is to consider a strain sensor configuration with the shoulder joint measured with a three-degree of freedom strain sensor. The strain sensor is able to detect the absolute angular displacement at the joint, so there is no need to map the measurement back to a global torso-centered system. In the strain sensor configuration, the measurement between the torso and the limbs are disjointed and prevent propagation of error from the torso to the limb segments. The measurement error is computed with error applied to the torso component (Figure 3.2. This simple experiment illustrates the sensitivity of the system to torso measurement error. There are two important aspects of the error accumulation in the limb-segments. First, as the system is rotated to the rotation invariant representation, the error on the torso is propagated down the limb segments. The rotation invariant representation removes the environmental rotation. Second, the additional error is strictly a linear

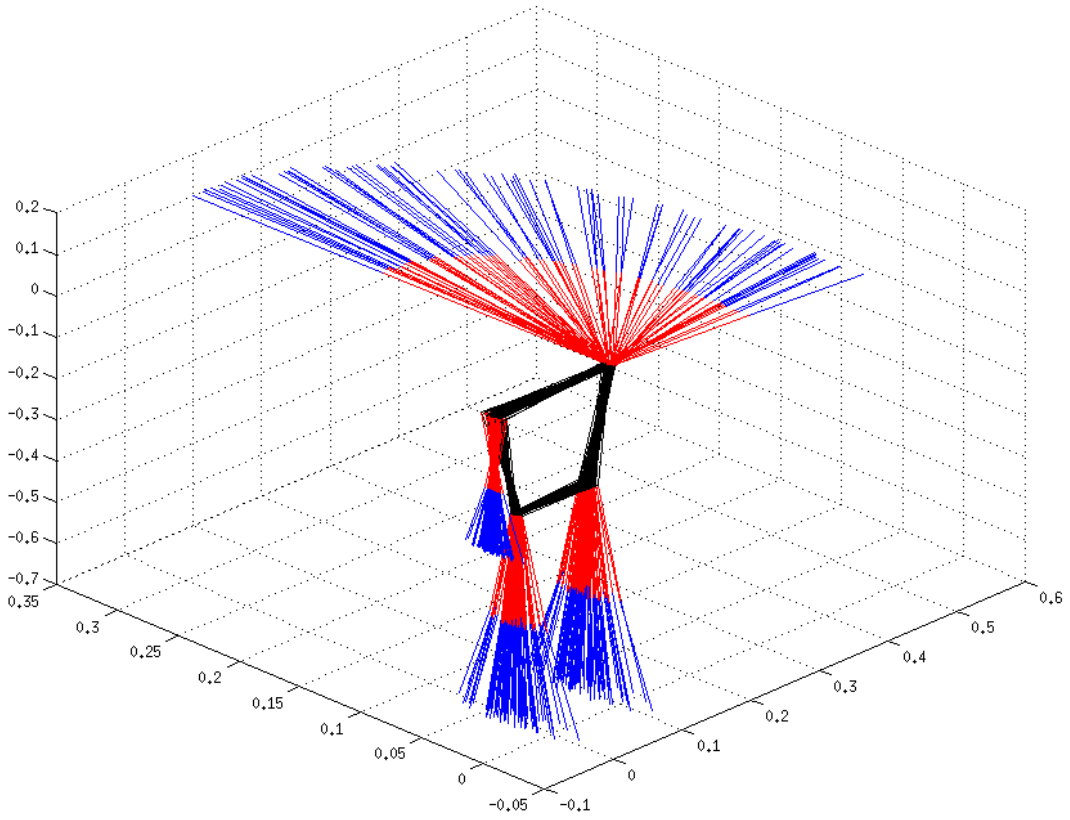
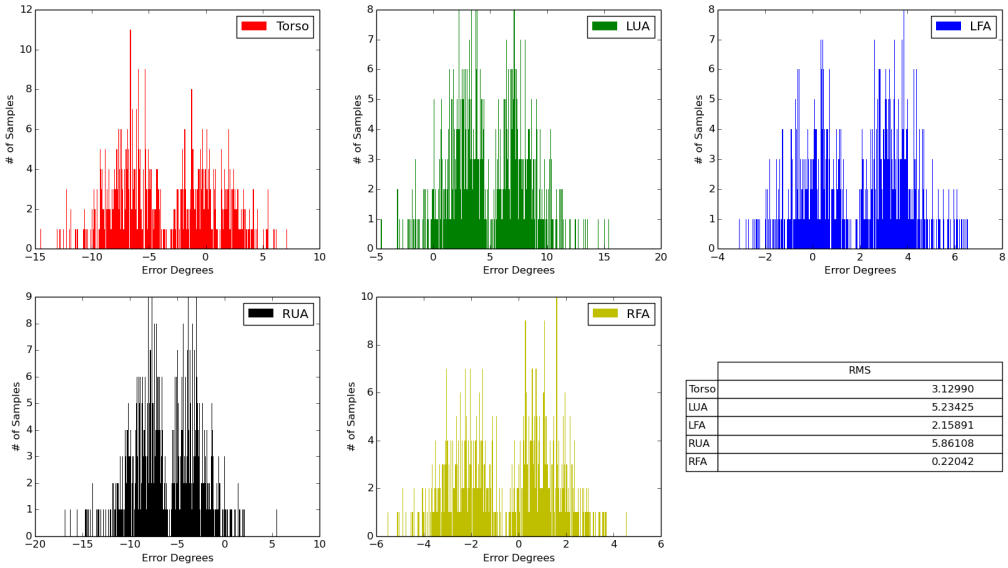



Figure 3.1: Synthetic Arm Sweep Activity with $\pm 5^{\circ}$ error applied to each component

combination of this error applied to the subsequent limbs in the body-model. Focusing on the first point, this propagation indicates that the system is sensitive to measurement error within the torso segment. This sensitivity is important as it limits the achievable measurement accuracy. The error is compounded in a body-worn system when the torso sensor moves due to garment shift during an activity. Mitigation of this effect is discussed in Section 3.2. The second point illustrates that the error does build within the system. This additive error effect will be important in the subsequent sections when the overlap of the activity space is examined to provide an expected precision evaluation of an noisier system. This evaluation adds to the previous work done in Lewis (2011) by quantifying the error with respect to the body-model definition. Additionally, as the body-model is defined, the error propagation posed by the combination of the body-worn sensors creates a challenge when classifying activities. Section 3.2 investigates the source of underlying error and methods to mitigate the contribution at the sensor and body-level domain.




| | RMS |
|---|---|
| Torso | 3.12990 |
| LUA | 5.23425 |
| LFA | 2.15891 |
| RUA | 5.86108 |
| RFA | 0.22042 |

Figure 3.2: Gaussian error on only the torso bounded to $\pm 10^{\circ}$

## 3.2 Analysis of Error

This section details the underlying evaluation of the current jumpsuit system V2. The focus begins by exploring the system as a sensor fusion problem, with each sensor in consideration, the contribution to the system noise and error is considered. The quaternion representation and error/noise characteristics are also examined.

Additionally, filter investigation examines the underlying assumptions of each filter and explores the benefits and issues resulting from implementing each. The choice of filter is arrived upon within these considerations. The three filters investigated include a simple low-pass Finite Impulse Response (FIR), a spherical linear interpolation (SLERP) method, and a SLERP driven minimum bounding circle approach. Necessary filter design considerations are provided, and the assumptions of each filter are explored. These investigations, combined with the error and noise characteristics of the sensor systems, resulted in the choice of the SLERP driven minimum bounding circle as a suitable filter.

### 3.2.1 Individual Component Error

To examine the individual component error the necessary understanding of bias, frequency perturbations and stability of the measurement is performed in the sensor domain. It is important to note that each

component in the system must be tuned separately to account for offsets in each sensor domain case. A static test assembly was constructed to account for and remove the bias and frequency perturbations of the underlying sensors.

**Static Testing**

In the static test setup the upper body sensors were affixed to the PVC upper body static assembly shown in Figure 3.3. This testing assembly allowed the underlying measurement error to be examined from the compass and gyro. Additionally, it provides a simple static measurement test bed to examine the filter contributions described in Section 3.2.5.

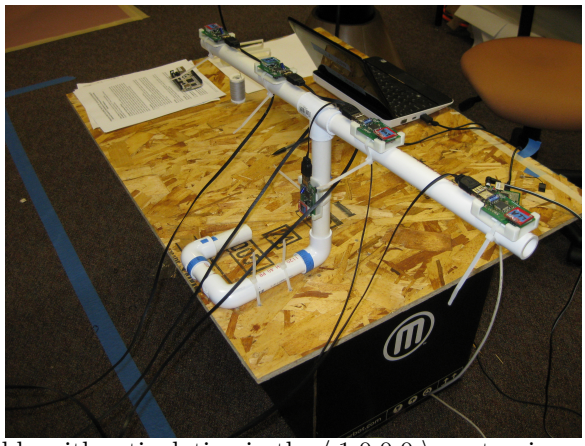


Figure 3.3: Static test assembly with articulation in the $\langle$ 1,0,0,0 $\rangle$ quaternion component for the upper body

### 3.2.2 Compass

The 3-axis Honeywell compass HMC6343 used, which has a max update rate of 10Hz, has two main calibration methods that must be completed to account for environmental offsets: including hard-iron calibration, and geo-location of the device. The hard-iron calibration allows the compass to mitigate effects of large stationary ferrous objects within the vicinity. Geo-location calibration is managed by accounting for the magnetic declination from true north with respect to the location of the device. These methods are detailed in the datasheet for the device by Honeywell. Compass error is examined in two stages: the raw angle error, and the quaternion error.

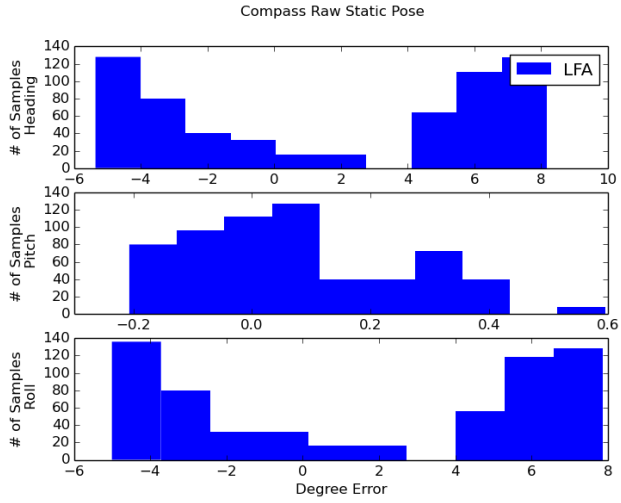
***Raw Angle Error***

Examintation of compass error is done in three static poses to produce a complete picture of the error in the environment. These poses consist of the compass at a level orientation, and $\pm 90^{\circ}$ pitch. The level

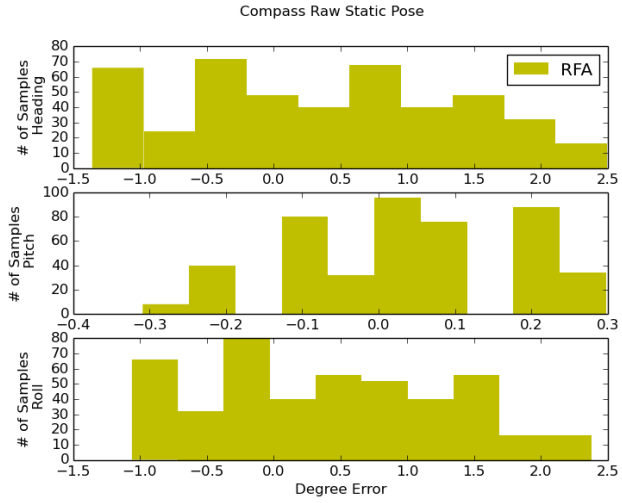
orientation yields the lowest RMS degree error contribution, while the ±90° pitch yields the worst RMS degree error (Table 3.1). It is important to understand that the RMS degree error does not always provide a complete picture. For example, consider the left forearm (**LFA**) segment across the ± 90 and level orientation. The RMS error is the highest compared to all the other segments, the underlying issue is that the distribution of the error for the **LFA** is significantly more peaked (Figure 3.4a compared to Figure 3.4b). This shows that the error within some segments is significant, even in a static position. The distribution of the error illustrates a dual peaked response on the heading and roll axis (Figure 3.4). The mixture of the distribution is important as it impacts the filtering considerations (Subsection 3.2.5).

Table 3.1: Average RMS degree error for static poses illustrating the difference between ±90° and level orientation of the compass

| Limb ID | 90 | Level | -90 |
|---|---|---|---|
| T | 0.57 | 0.57 | 1.045 |
| LUA | 1.185 | 0.137 | 0.695 |
| LFA | 1.47 | 1.1 | 1.7 |
| RUA | 0.915 | 0.582 | 0.565 |
| RFA | 0.655 | 0.29 | 0.5085 |




(a) Example of Left Fore Arm measurement noise during a static test at a pitch of 90° w.r.t the y-axis of the system



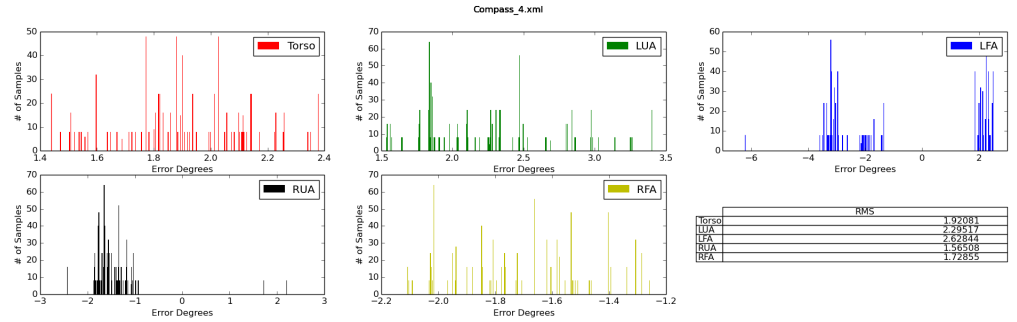

(b) Example of Right Fore Arm measurement noise during a static test at a pitch of 90° w.r.t the y-axis of the system

Figure 3.4: Static testing to examine the noise contributions of the underlying components of the compass in degree form

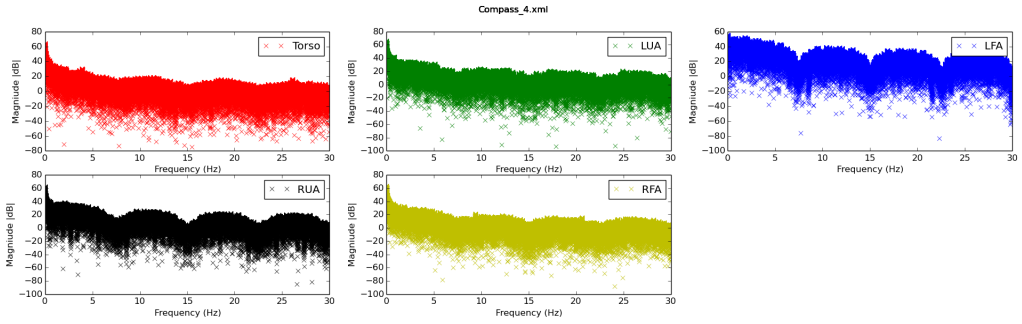
### *Quaternion Error*

Because the quaternion representation is built directly from the raw measurements of the compass, the quaternion error directly reflects underlying sensor error (Figure 3.5). This error is computed as a function of

the angle error between two unit vectors. The initial unit vector is rotated by the initial quaternion reading, and the error calculations are then computed to be the angle error between the initial unit vector and the unit vector rotation of the subsequent quaternion values. Comparing the raw compass error distribution in Figure 3.4a with the quaternion error on the same limb in Figure 3.5a shows that the peaks within the underlying compass data transition into the peaked perturbations of the quaternion error. Examining the frequency plot in Figure 3.5b, it becomes evident that there are no significant peaks in the frequency domain. Furthermore, the correlation between the measurement error, computed using the spearman correlation method, shows that the error is un-correlated, as shown in Table 3.2. This method was chosen because the distribution of the error is not guaranteed to be normal. Since the error is un-correlated, the system can be treated as individual components and filtered at the segment level.




(a) Example quaternion compass noise at the level orientation




(b) Frequency dB plot of the quaternion compass error

Figure 3.5: Static testing to examine the noise contributions of the underlying components in quaternion form

Mitigation of the error within the compass is investigated in sections 3.2.4, 3.2.5, and 3.2.9. These sections focus on the filtering and calibration considerations.

Table 3.2: Error Correlation between the quaternion representation of the limbs

| Limb ID | Torso | LUA | LFA | RUA | RFA |
|---|---|---|---|---|---|
| Torso | 1. | 0.11 | -0.12 | 0.07 | 0.01 |
| Left Upper Arm | 0.11 | 1. | -0.16 | 0.12 | 0.20 |
| Left Forearm | -0.12 | -0.16 | 1. | -0.03 | 0.10 |
| Right Upper Arm | 0.07 | 0.12 | -0.04 | 1. | 0.10 |
| Right Forearm | 0.013 | 0.20 | 0.10 | 0.10 | 1. |

### 3.2.3 Gyro

The two 2-axis gyros on each board are able to generate a 3-axis angular rate of rotation about the locally defined board axis, which is used to generate the relative additional rotation between the compass updates. These gyros must be calibrated to function properly, which is accomplished by finding an appropriate offset for each gyro in a stable state. This offset is an averaged value, as there is some fluctuation within the gyro readings. The compass and gyro update are then combined in an equal weighting, and the gyro output is put through a low-pass filter before the measurements are incorporated into the quaternion update cycle. The low-pass filter selection for this is a 3-point moving average filter to mitigate high-frequency noise components. This filter was chosen to facilitate a quick measurement and thus minimize extra delay that could occur with a longer filter at this level.

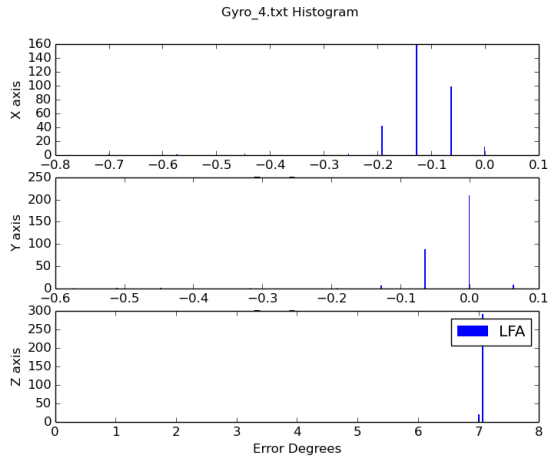
***Raw Angle Error***

Examination of the gyro error is done for the un-calibrated and calibrated measurements.Figure 3.6 shows un-calibrated and calibrated steady state error in the gyros. The error is significantly reduced in the calibrated data, however, it still contains some amount of bias. Figure 3.7 shows the noise before and after the calibration on the right upper arm segment. The difference in noise distributions is important because each component will contribute in un-correlated ways to the overall error of the body-model. It is important to note that even after a static calibration is applied to each sensor to counter the bias in the gyros, the error is not completely mitigated.

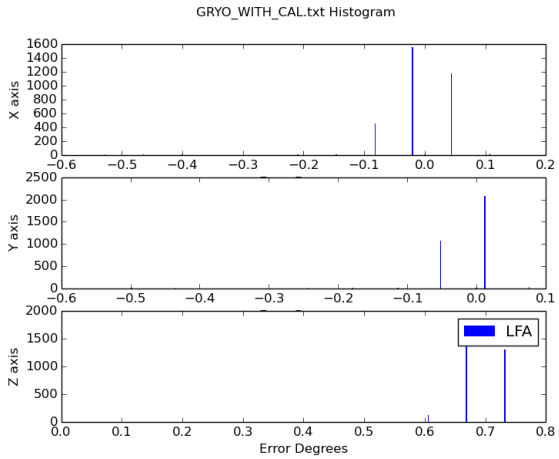
***Quaternion Error***

$$Q_{Out}(t) = \begin{cases} Q_{Compass}(t-1) * Q_{Gyro}(t) & \text{if previous update was compass} \\ Q_{Out}(t-1) * Q_{Gyro}(t) \end{cases} \tag{3.1}$$

Equation 3.1 combines the quaternions from the compass or previous quaternion with the updated gyro quaternion. Equal weight on the quaternions allows the gyro to contribute fully at each update event.

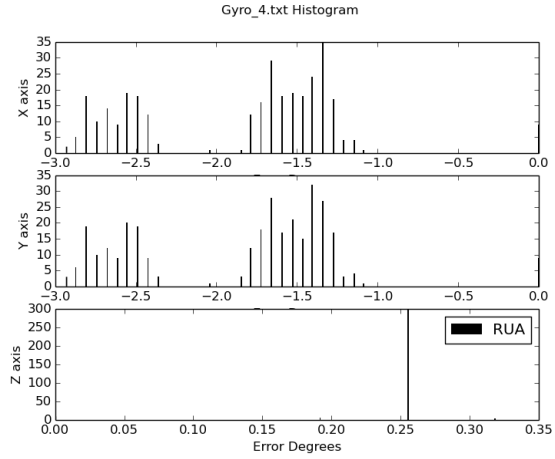



(a) Example gyro error in a steady state for the Left Forearm

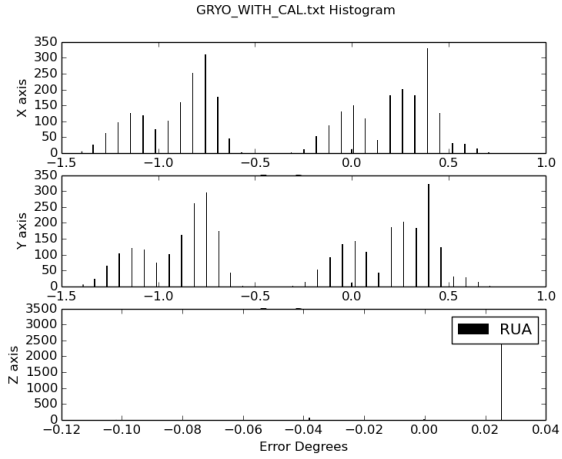



(b) Example gyro error in a steady state after the online calibration

Figure 3.6: Static testing to examine the noise contributions of the underlying components and the calibration correction




(a) Example gyro error in a steady state for the Right Upper Arm segment




(b) Example gyro error in a steady state with calibration correction

Figure 3.7: Static testing to illustrate different noise distributions for different limb segments before and after calibration

However, this approach can be problematic as the gyros are prone to integration drift over time. Therefore, at the compass update event, the local quaternion is fully defined by the compass quaternion. This design decision mitigates the integration error at the cost of trusting the compass update. Section 3.2.5 investigates methods to produce a more accurate quaternion representation through system level filtering.

### 3.2.4 Online Calibration

Calibration is performed on the suit to mitigate gyro and compass noise issues. During the initial few seconds of power-up, a simple running average is computed on the gyro values. These values are stored and used to adjust the bias within the gyros during the operation of the system. The subject is required to maintain a steady position to allow successful calibration of the gyros online; the code base that runs on the tier 2 node, `MotionCapture`, provides this functionality.

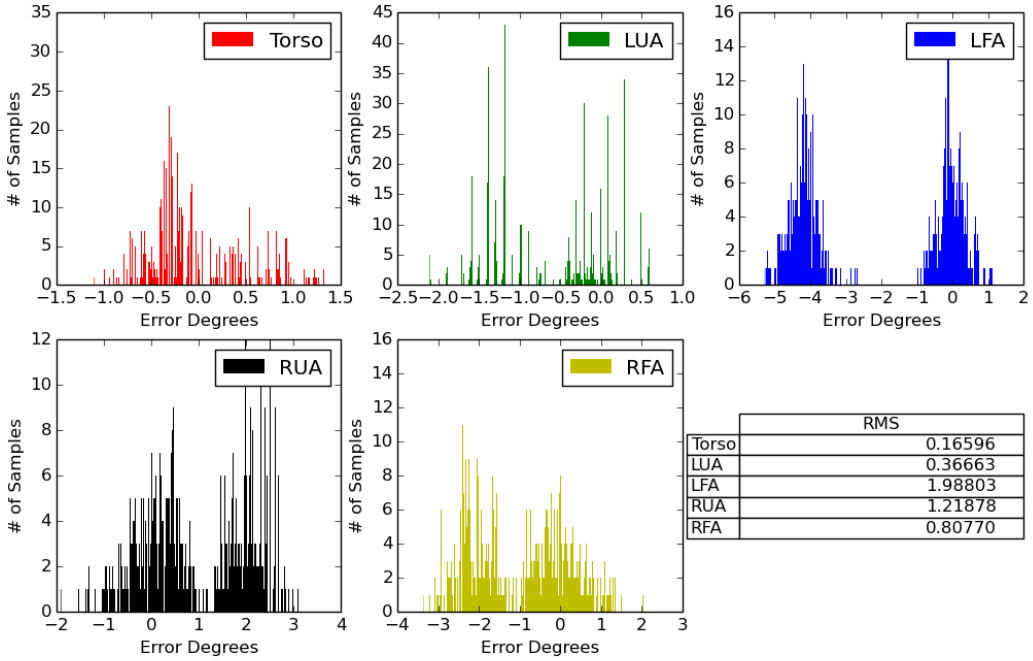
### 3.2.5 Filter Design

Sections 3.2.2 and 3.2.3 examined the underlying errors in the compass and gyro components that are combined to form the quaternion representation of the body-model presented in Chapter 2.3. Additionally, it was shown in Table 3.2 that the error is uncorrelated, which allows the consideration of the limbs as separate entities. To mitigate the error contributions, the investigation of a low-pass Finite Impulse Response (FIR), weighted Spherical Linear Interpolation (SLERP), and a SLERP driven minimum bounding circle are explored.

The activity domain of the filter characteristics in consideration consists of activities within the 1-2Hz range, which can be related to the angular rate of motion ($2\pi/s$ to $4\pi/s$). Combining the range of angular rates of rotation with the 60 Hz sampling rate, the expected maximum angular rate during a sample should be no more then approximately $\pi/18$ rad/sample (10 deg/sample ). This restriction of the activity space with respect to the frequencies of interest allows the consideration of lower-order filters.

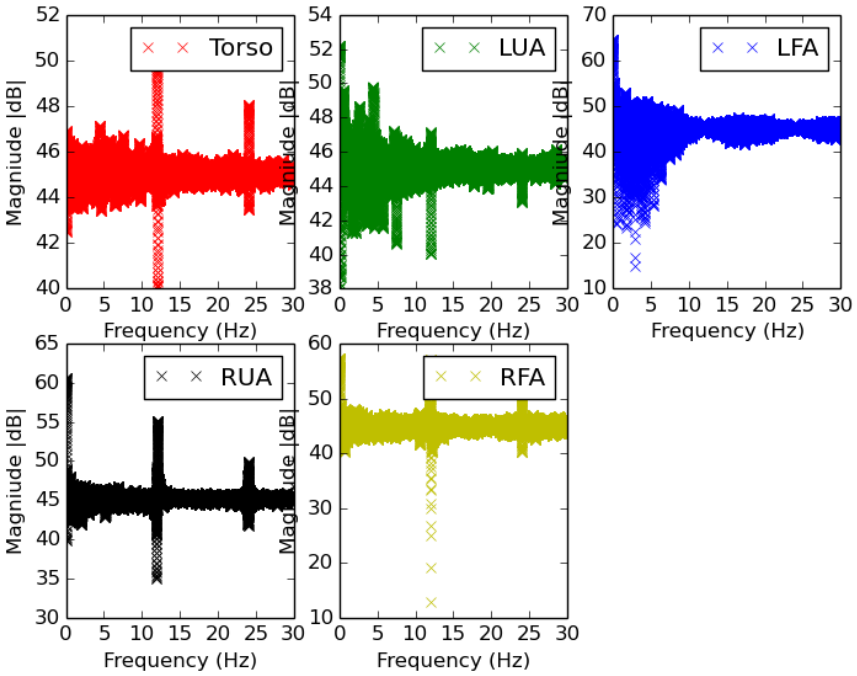
The same set of static tests is used in the comparison of the filters. Figure 3.8 shows the test case to which the filter in each sub-section will be compared. An important aspect of Figure 3.8a is that the Left Fore Arm has a peaked response quite different from the other segments. This peaked response exists because of the compass, as was shown in Figure 3.4a. Figure 3.8b shows the frequency response of the static measurement of the linked kinematic model. The two peaks at 12Hz and 24Hz form a high frequency component in comparison to the speed at which most of the activities take place.

Using the synthetic example arm sweep activity, the following examines this filter as applied to a uniformly noised example. Using the arm sweep activity, the error can be computed using the ideal synthetic activity. This activity allows comparison of the different filters, which is useful if a filter is suspected of under or overshooting a transition point and causing the activity to fail classification.




| | RMS |
|---|---|
| Torso | 0.16596 |
| LUA | 0.36663 |
| LFA | 1.98803 |
| RUA | 1.21878 |
| RFA | 0.80770 |

(a) Example level static measurement before filtering




(b) Example level static measurement before filtering frequency plot

Figure 3.8: Example static baseline measurement

### 3.2.6 Low-Pass FIR

Based on the simplifying assumption limiting the angular rate of the activities, a 5 Hz low-pass Finite Impulse Response (FIR) filter is considered in this section. This filter, designed in MATLAB, has a nominal 0.1 dB ripple in the pass band and is composed of 60 coefficients (Figure 3.9). This section illustrates the filter characteristics and explores its impact on the static test. With adaptation of the synthetic activity from Section 3.1, the system is built to define the known and arbitrary noise contributions to the system in order to explore the filter effectiveness using the frequency response and corresponding noise distribution.

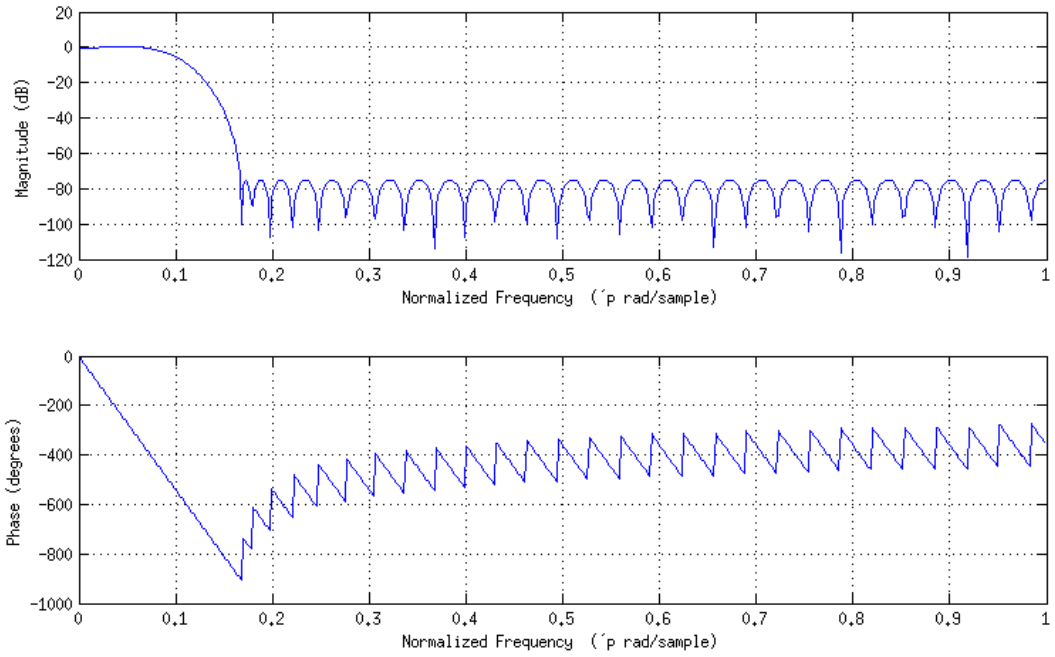
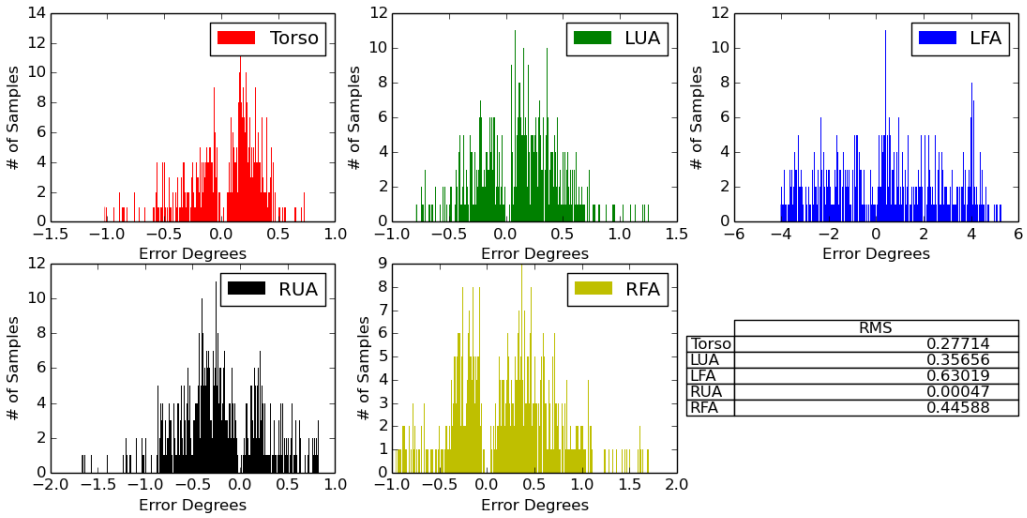



Figure 3.9: Low-Pass Filter characteristic

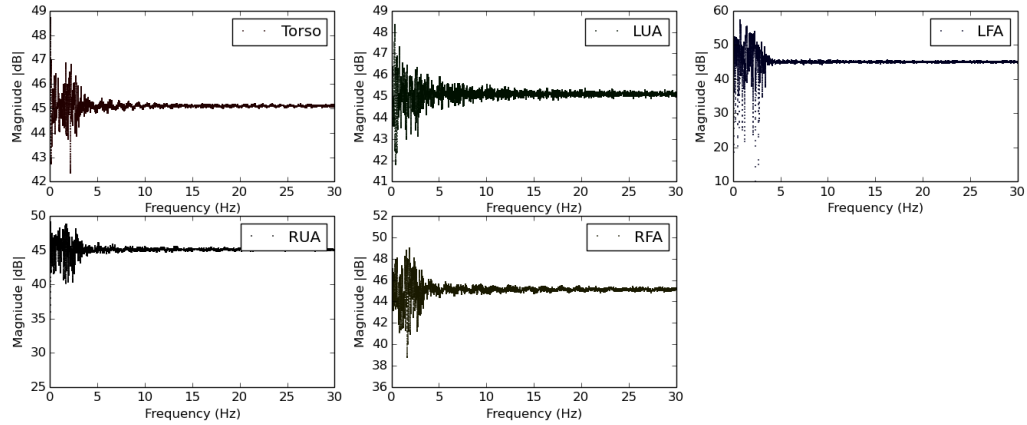
From the previous baseline measurements (Figure 3.8), the frequency perturbation at 10Hz-12Hz is mitigated (Figure 3.10). The distribution shown in Figure 3.10a shifts to become a better approximation of a normal Gaussian distribution.

Shifting from the static examination of the low-pass FIR, the synthetic Arm Sweep activity is examined. To account for delay of the FIR filter, the samples were shifted to prevent an erroneous error calculation. The noise in the synthetic example is a uniformly sampled non-correlated degree error on the **X,Y,Z** axis of rotation. The error is used to compute a rotation quaternion for each limb, which is then applied to the original quaternion to generate the error in the sequence.




| | RMS |
|---|---|
| Torso | 0.27714 |
| LUA | 0.35656 |
| LFA | 0.63019 |
| RUA | 0.00047 |
| RFA | 0.44588 |

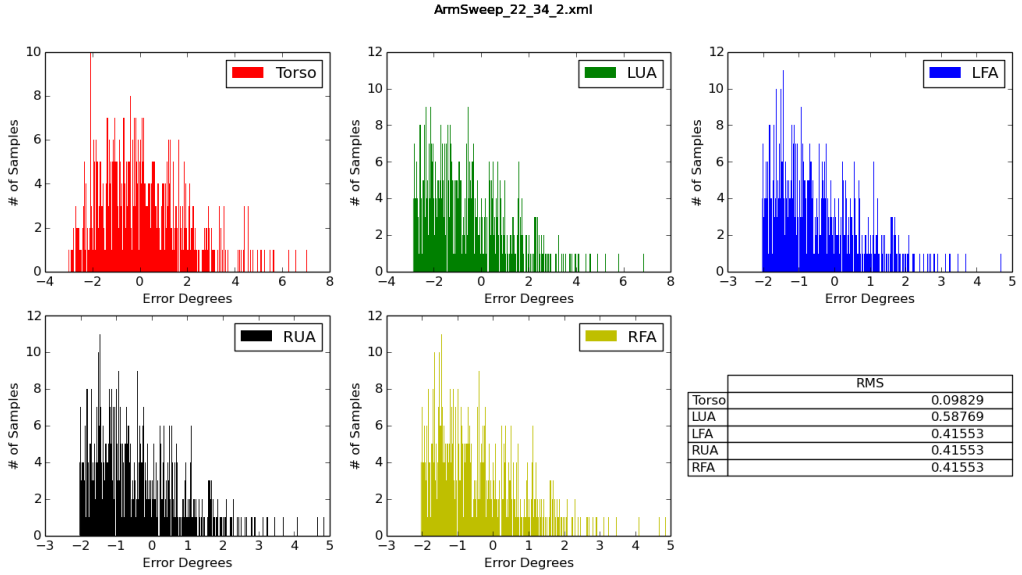
(a) Example level static measurement after the low pass filter




(b) Example level static measurement after the low pass filter frequency plot

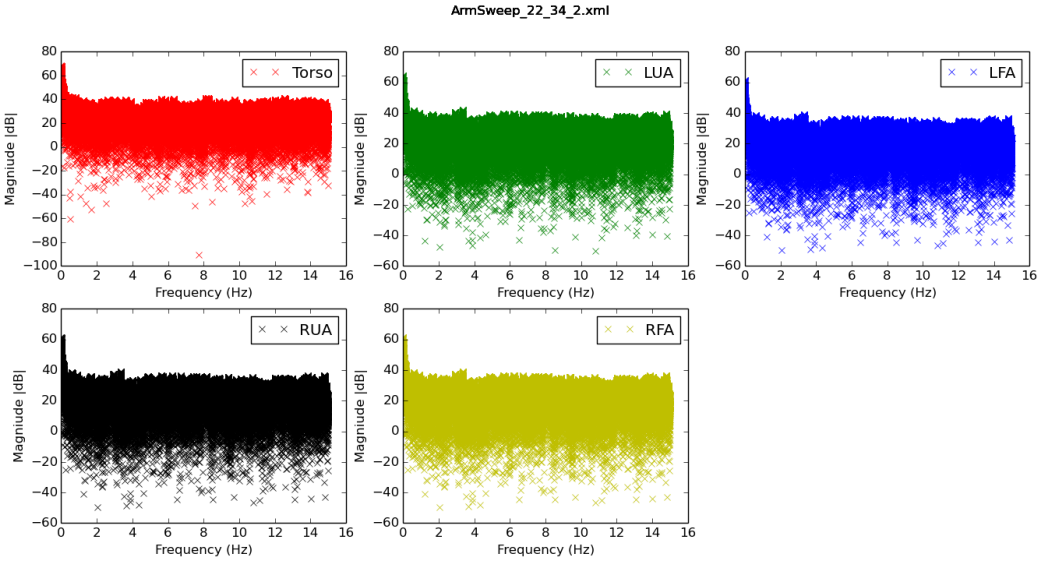
Figure 3.10: Performance of the Low pass filter on the static error measurement

While the low-pass FIR filter reduces the overall high-frequency components of the static and the dynamic synthetic testing, the two SLERP methods have also been investigated to examine other methods that would potentially extend beyond the restricted activity set considered here.

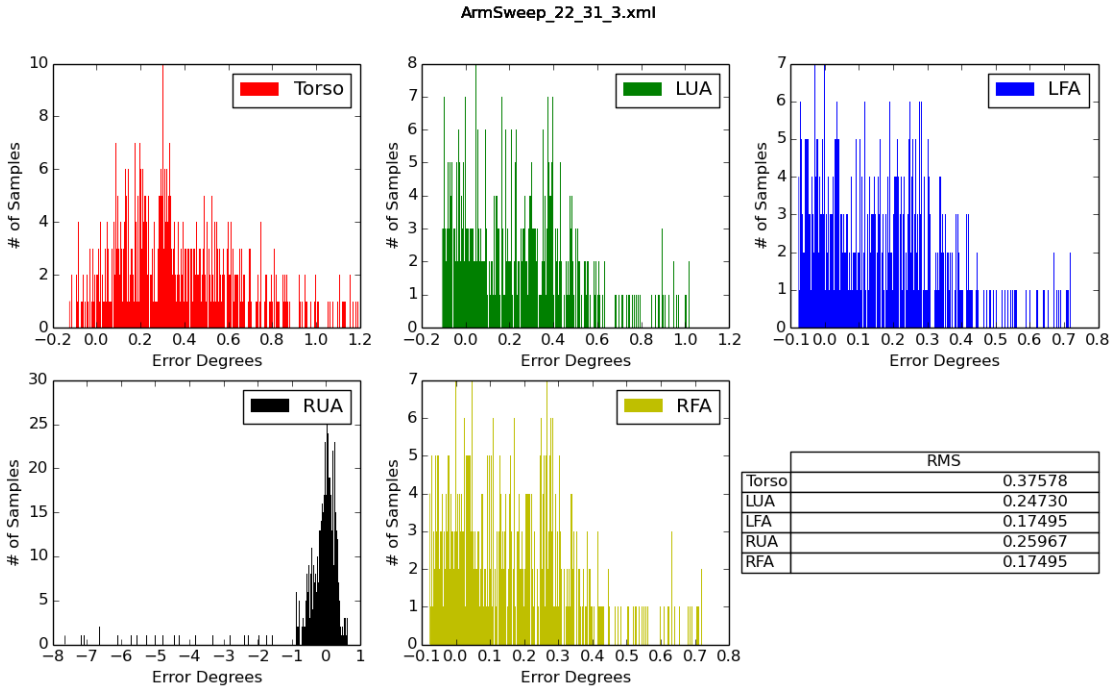



| | RMS |
|---|---|
| Torso | 0.09829 |
| LUA | 0.58769 |
| LFA | 0.41553 |
| RUA | 0.41553 |
| RFA | 0.41553 |

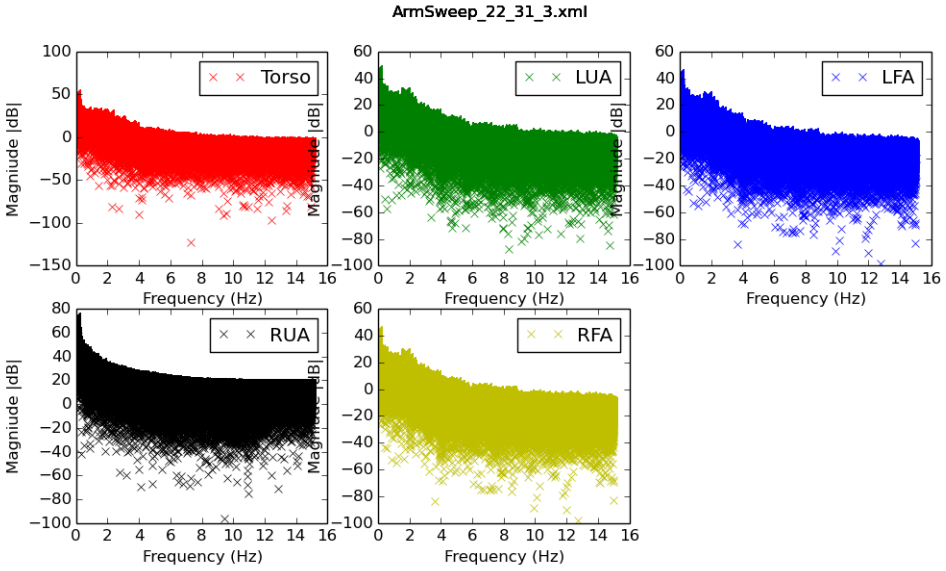
(a) Example dynamic measurement before filtering




(b) Example dynamic measurement before filtering frequency plot




(a) Example dynamic measurement after filtering




(b) Example dynamic measurement before filtering frequency plot

Figure 3.12: Baseline error measurement of the noised ArmSweep activity

### 3.2.7 SLERP

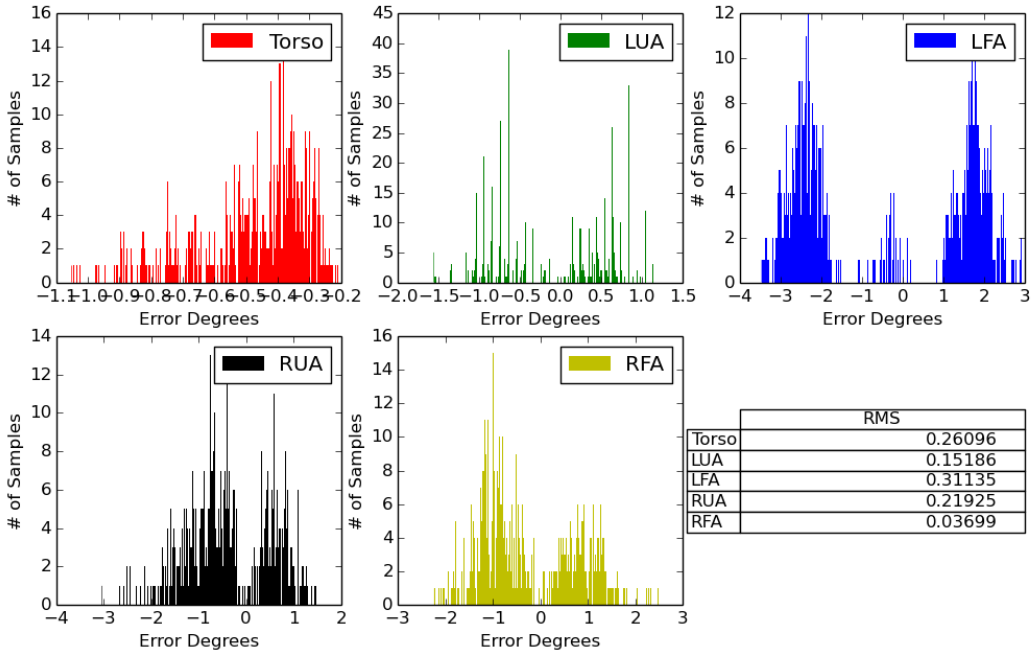
Spherical Linear Interpolation is a method that can be used to smooth curves by computing the angle difference based on a factor **t** (Han (2006)). This value effects weighting the new quaternions contribution to an update. The effect of SLERP is a low pass filter as a function of **t** and angle $\Omega$.

$$Slerp\left(q_1, q_2, t\right) = \frac{\sin\left[(1-t) * \Omega\right]}{\sin\left(\Omega\right)} * q_1 + \frac{\sin\left[t * \Omega\right]}{\sin\left(\Omega\right)} * q_2$$
$$\Omega = 2 * \arccos\left(q_1 \circ q_2\right) \tag{3.2}$$
$$\circ \text{ represents the quaternion dot product}$$

A **t** value of 0.05 was chosen for the Torso component and a value of 0.5 was chosen for the limb segments. It is important to consider the frequency plot in Figure 3.10b. There are still components at the "high" frequency in the limb segments that will create measurement error. The choice of the 0.5 factor on the limbs allows the segments to quickly track changes in the orientation of the limbs. This approach is in contrast to the torso segment, which was restricted on the assumption that the body as a whole would not move quickly within the current activity domain.

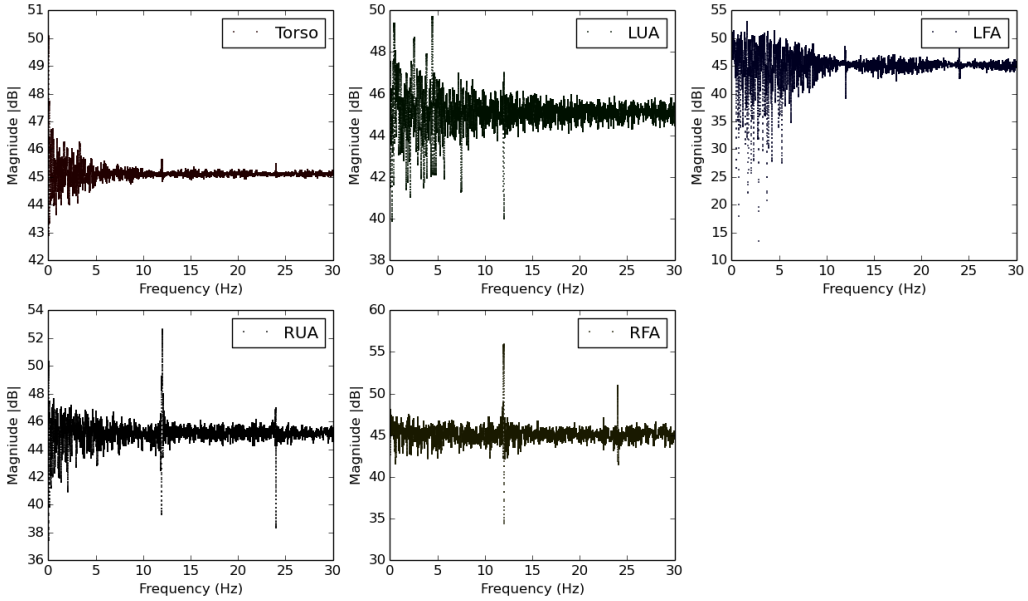
Using the SLERP algorithm on the Arm Sweep Activity with a **t** of 0.5 for the segments in the body model the noise plots on the system were examined. Figure 3.14 shows the dynamic test with SLERP applied. Examining the frequency plot in Figure 3.13b one notices that the "high" frequency components at 10 and 12 Hz are not removed in comparison to the low-pass FIR filter result in Figure 3.10b. However, the benefit to the SLERP filter method is that it doesn't require a long set of filter coefficients to compute the current output thereby possibly reducing computation costs. The reduction would be a function of the system specifics though and the trade-off between computing a trig function verses a set of multiplication and addition computations.

The tracking of the system is shown in Figure 3.15, where the system is supplied with an "impulse" quaternion. The impulse quaternion is a rotation from time **t=0** to **t=1** of 90°. After **t=1** the segment is steady with no rotation. The figure illustrates how the SLERP function interpolates and when it reaches a steady state. For increasing values of t up to 0.5, the tracking of the limbs become quicker as noted by the number of samples it takes for the system to reach a value of 1.




| | RMS |
|---|---|
| Torso | 0.26096 |
| LUA | 0.15186 |
| LFA | 0.31135 |
| RUA | 0.21925 |
| RFA | 0.03699 |

(a) Example of the noise distribution of the level static measurement after the weighted SLERP filter is applied




(b) Example of the level static measurement after the weighted SLERP filter is applied in the frequency domain

Figure 3.13: Example of the level static measurement after the weighted SLERP filter is applied

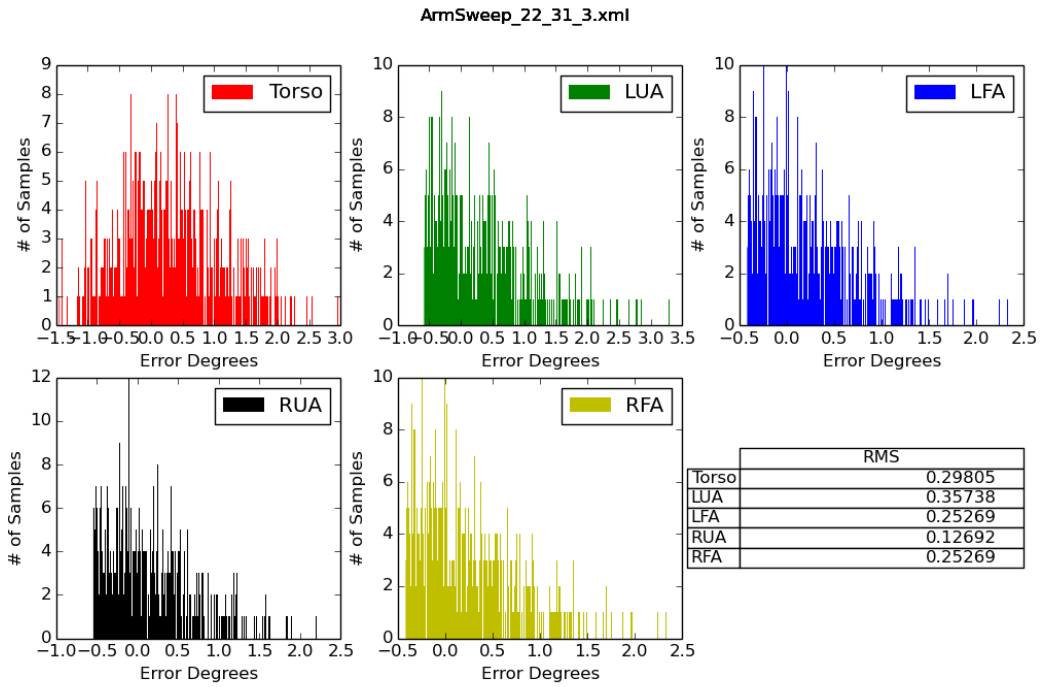



Figure 3.14: Example of synthetic ArmSweep activity dynamic error with SLERP applied

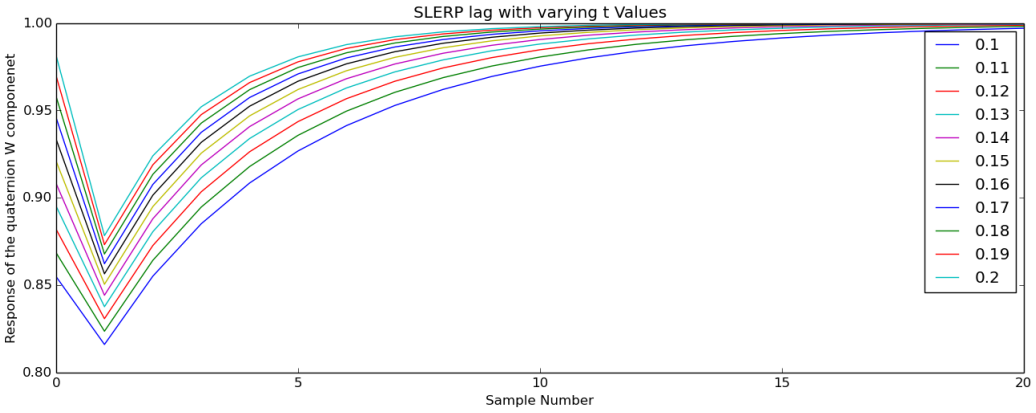



Figure 3.15: Example of the sample lag with varying t values

### 3.2.8 Minimum Bounding SLERP

This section explores the effects of the minimum bounding circle algorithm with SLERP (Algorithm 3). The SLERP driven minimum bounding algorithm is applied to the static tests and dynamic tests. Figures 3.16a and 3.16b show the algorithm applied to one of the static tests. The important aspect of the frequency plot compared to the previous Figure 3.13b is that the peaks in the frequency domain at 12 and 24 Hz have

been removed. The benefit to the minimum bounding circle is that the frequency range of movement is not bounded as is the case in the low-pass FIR filter.

A comparison of the filters is given in Table 3.3. This comparison, an average over 10 sequences, is made between the synthetic dynamic tests. With the same error applied to each synthetic test across the filters, it becomes apparent that the best performing filter in the set is the minimum bounding circle algorithm (Section 3.2.8).

**Algorithm 3** SLERP driven minimum bounding circle

QSet is the group of quaternions that the algorithm optimizes on
QSet $\geq 3$
$qOld \leftarrow QSet[0]$
$t \leftarrow 0.5$
$EArray \leftarrow 0$

**for** i:=0 to len(QSet) **do**
    $EArray[i] \leftarrow quatDist(qOld, QSet[i])$
**end for**
$maxIndx \leftarrow MAX(EArray)$
$qNew \leftarrow quatSlerp(qOld, QSet[maxIndx], t)$

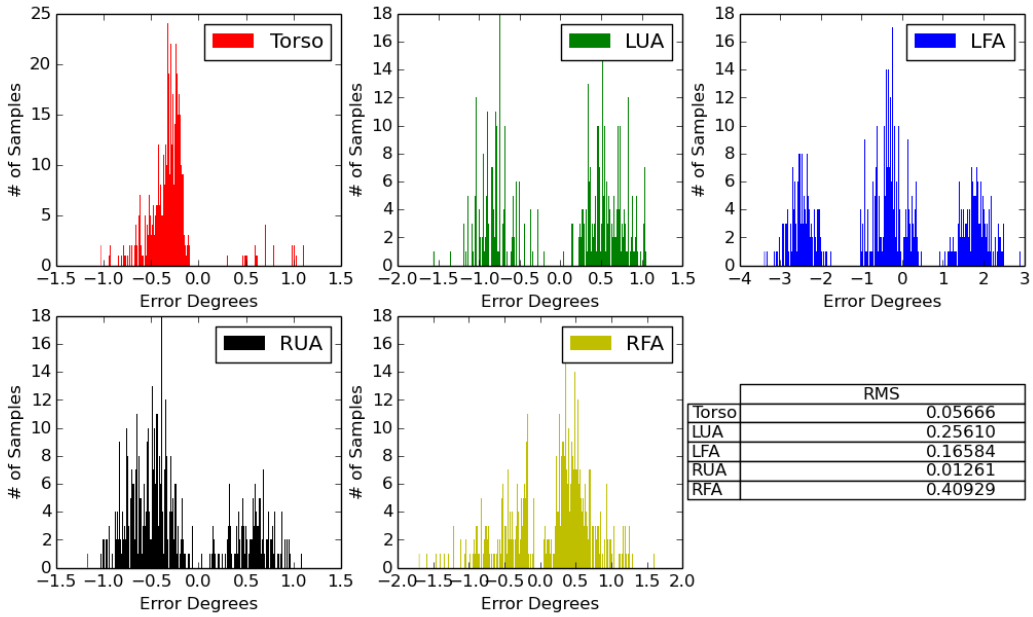
Gradient Decent
$t \leftarrow 0.1$

**while** count$\leq$ breakOut **and** err $<$ Thresh **do**
    **for** i:=0 to len(QSet) **do**
      $EArray[i] \leftarrow quatDist(qNew, QSet[i])$
    **end for**
$maxIndx \leftarrow MAX(EArray)$
$qTemp \leftarrow quatSlerp(qNew, QSet[maxIndx], t)$
$qOld \leftarrow qNew$
$qNew \leftarrow qTemp$
$err \leftarrow quatDist(qOld, qNew)$
$count \leftarrow count + 1$

**end while**

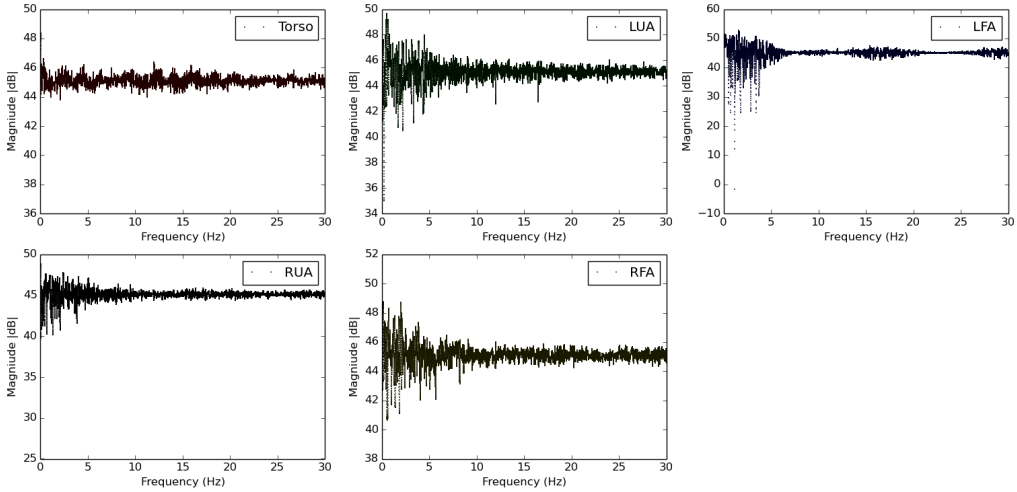
Table 3.3: Average RMS Degree Error of the upper body limbs for the FIR, SLERP, and Minimum Bounding Circle filters

| Filter | Avg. RMS |
|---|---|
| Min. Bounding Circle | 0.165 |
| Low-Pass FIR | 0.173 |
| SLERP | 0.303 |




| | RMS |
|---|---|
| Torso | 0.05666 |
| LUA | 0.25610 |
| LFA | 0.16584 |
| RUA | 0.01261 |
| RFA | 0.40929 |

(a) Example level static measurement after the minimum bounding circle




(b) Example frequency spectrum of the static tests with the minimum bounded circle algorithm applied

Figure 3.16: Example of static testing with the minimum bounding circle algorithm applied

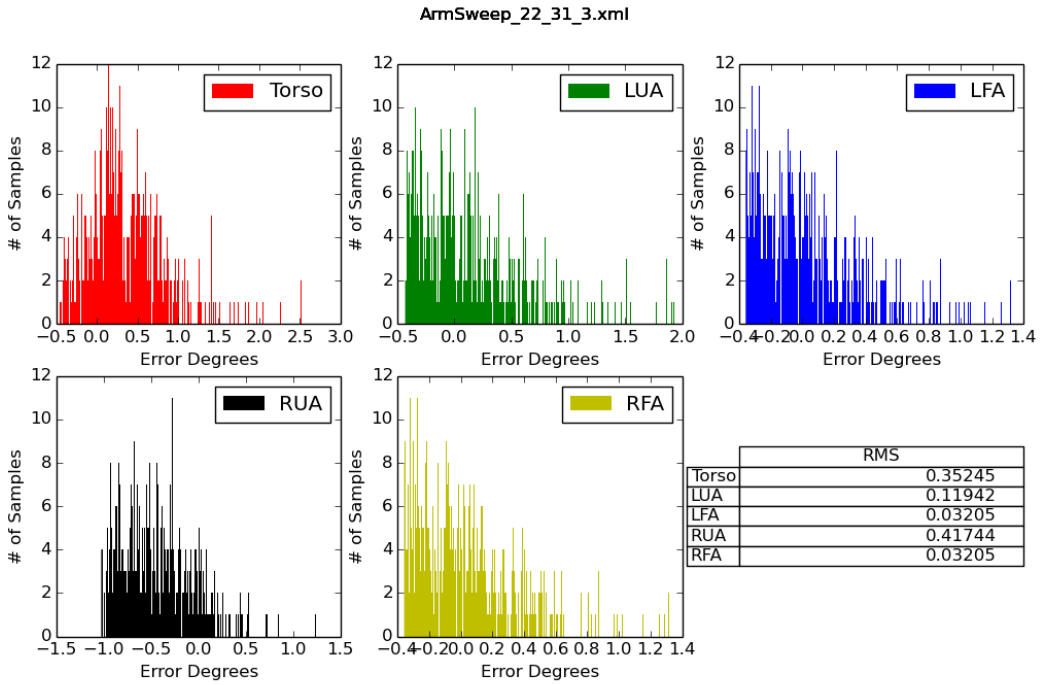



Figure 3.17: Example of synthetic dynamic error with the minimum bounding circle algorithm applied

### 3.2.9 Calibration Considerations

Calibration of the system is necessary to remove donning/doffing offset effects of the system and is set up to minimize user interruptions. It is desired and enforced that the calibration is a one-time event for each subject when the jumpsuit is worn for the testing. At present, the calibration is collected and applied off-line, however, in a more user-friendly system the calibration would be integrated into the sensor domain online-bias correction calibration, (Section 3.2.4). Calibration is accomplished at the system level so that the necessary filtering and bias correction is completed for the best quaternion representation possible. The calibration consists of two static poses: sitting and standing. Specifically, the quaternion pose representation is given, in the un-normalized form, in Table 3.4 for the two poses.

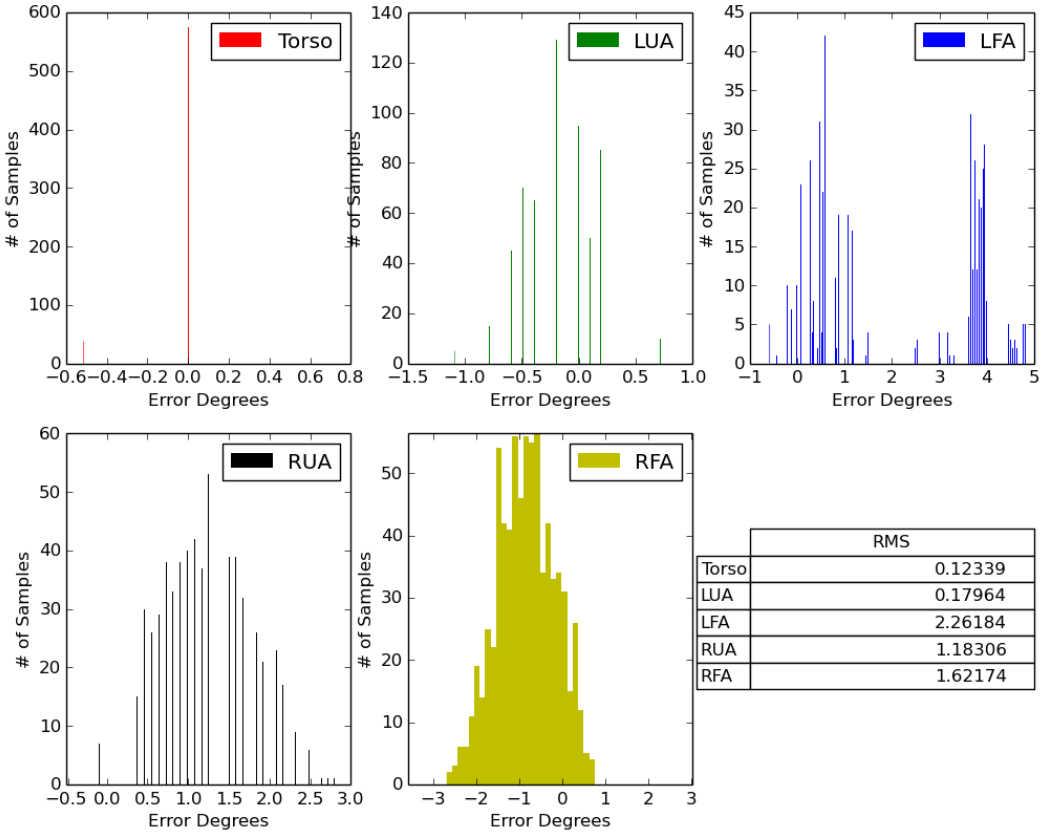
Table 3.4: Calibration Poses

| Limb Identifier#1 | Sitting #2 | Standing |
|---|---|---|
| Torso | ⟨ 1,0,0,0 ⟩ | ⟨ 1,0,0,0 ⟩ |
| Left Upper Arm | ⟨ 1,0,-1,0 ⟩ | ⟨ 1,0,-1,0 ⟩ |
| Left Forearm | ⟨ 1,0,0,-1 ⟩ | ⟨ 1,0,0,0 ⟩ |
| Right Upper Arm | ⟨ 1,0,-1,0 ⟩ | ⟨ 1,0,-1,0 ⟩ |
| Right Forearm | ⟨ 1,0,0,-1 ⟩ | ⟨ 1,0,0,0 ⟩ |
| Left Thigh | ⟨ 1,0,0,-1 ⟩ | ⟨ 1,0,-1,0 ⟩ |
| Left Shin | ⟨ 1,0,-1,0 ⟩ | ⟨ 1,0,0,0 ⟩ |
| Right Thigh | ⟨ 1,0,0,-1 ⟩ | ⟨ 1,0,-1,0 ⟩ |
| Right Shin | ⟨ 1,0,-1,0 ⟩ | ⟨ 1,0,0,0 ⟩ |

It is important to note that the left and right sides of the body are equal, which is a design decision that allows the poses to be effectively mirrored from right to left. To accomplish the calibration for an individual the suit is donned and an initial data collection is accomplished. The standing pose is such that the individual is standing straight with feet together and arms by their side. The sitting pose calibration is accomplished with the individual sitting in a chair with their feet flat on the floor, knees bent at a 90° rotation, and back straight up and down. The upper arms are placed against the side of the individual, and the forearm is at a 90° rotation with the hands in front of the torso, which mimics the calibration pose. The calibration is done to mitigate the effects of donning / doffing of the system and corrects for issues relating to fit of the garment and sensor shift. It is important to note that the calibration assumes that the sensors do not shift during an activity. Therefore, when a sensor shift does occur, the system may exacerbate the offset of the sensor due to the calibration. Additionally, calibration accounts for user anthropometric variations in the torso sensor due to gender. The calibration is completed for both sitting and standing before testing of the activities occurs. It is allowable to have the calibration correction done on the sitting or standing activities themselves, however, this may create un-intended offsets if the user is not sufficiently close to the

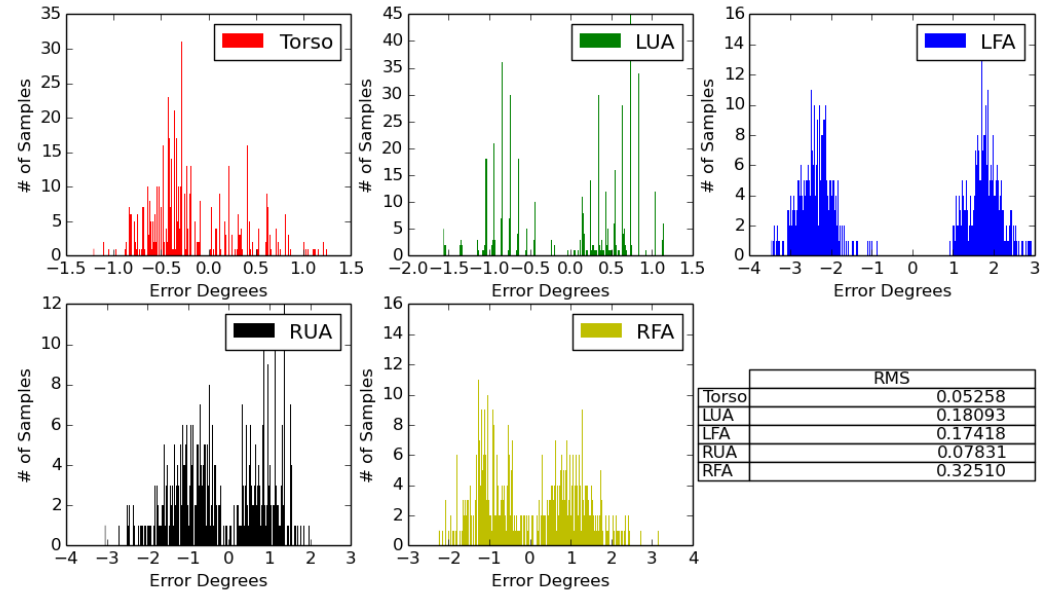
calibration pose. Therefore, it is advised that the user performs the calibration pose as correctly as possible.

Based on the standing calibration, the Figures 3.18a and 3.18b show the effect of calibration on the calibration pose. The error is computed between two unit vectors rotated by the respective quaternions. The RMS degree error in post-calibration (Figure 3.18b) versus pre-calibration (Figure 3.18a) is only slightly better. The small improvement is hypothesized to be because the system takes an average calibration reading at the pose level and applies it over the varying poses, which may or may not have sensor shift. While the calibration may not have a large effect on the RMS degree error in the Figure 3.18b, the benefit can be shown in the Left Fore Arm subplot of Figure 3.18a, where the error is now centered around 0. The error shift is in contrast to the previous Left Fore Arm un-calibrated measurement given in Figure 3.18a, where the error was biased about an offset of 2 degrees.




| | RMS |
|---|---|
| Torso | 0.12339 |
| LUA | 0.17964 |
| LFA | 2.26184 |
| RUA | 1.18306 |
| RFA | 1.62174 |

(a) Static measurement error for the calibration standing pose




| | RMS |
|---|---|
| Torso | 0.05258 |
| LUA | 0.18093 |
| LFA | 0.17418 |
| RUA | 0.07831 |
| RFA | 0.32510 |

(b) Static measurement error of the calibrated standing pose after calibration



Figure 3.18: Error before and after calibration

## 3.3 Classifier Investigation

Section 3.2 investigated the system on the sensor domain level, illuminated sources of measurement error, and provided methods to mitigate these errors. The focus now shifts from the low-level consideration of the system to that of the classifier and the activity domain, where activity space is examined within the context of overlap between the activities. This consideration provides a quantitative measure of how similar activities are within the activity space defined by the body-model. As described Section 3.2, the body-worn system contains a greater amount of measurement and noise error than a visual system. Threshold adjustment on the match between a known and observed activity can account for the additional measurement error.

### 3.3.1 Activity Overlap

It is desirable to understand how much overlap exists between a set of activities because overlap increases the chance of inter-class confusion. Circle intersection on the activity alphabet is considered to determine the overlap, which is computed as the minimum intersection over the set of segments in the body-model. This provides an examination into how close an activity is to another at its most distant point. Figure 3.19 provides a visualization of how the overlap is computed. Based on the weights of the activity, the radius of the circle varies and creates differing amounts of overlap. The area of the lens is where the confusion between two activities occur, and the distance from the center of the activity region to the circumference of the region is 1, due to the weights of the activities. The black line that bisects the confusion region represents the decision boundary, which is a curved line due to the weighted distance of the activity space.

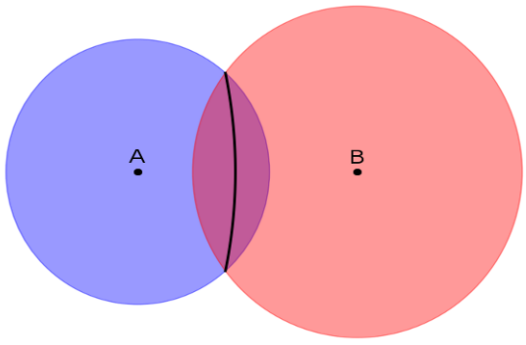



Figure 3.19: Example of the overlap computation. Points to the left of the black curve have a lower distance to Activity 'A' (blue) compared to Activity 'B' (red).

The activity overlap method shows that for given activities which are subsets of other activities (e.g., left-step / walking) there exists a significant overlap in the activity space. Table 3.5 shows the overlap

computed on a subset of the activities, which is the normalized minimum circle intersection. Considering the the two activities **Left-Step** and **Walk** in Table 3.5, the overlap between the activities is 33.3%. The overlap is the minimum overlap between all the segments across the two activity sequences an represents similar activities with a rating close to 1 and dissimilar activities toward 0. Using the activity overlap, one can additionally predict the classification accuracy of a set of activities. It is important to understand the assumptions that drive this prediction:

*Assumption 1:* The distribution of observations within the activity space are uniform and non-peaked.

*Assumption 2:* The overlap method identifies the minimum overlap between two activities. Coupled with assumption 1, this provides a lower bounds on the probability of misclassification between two activities.

The assumptions allow the presentation of the misclassification probability within the confused area, and as an extension, the expected accuracy given the activity set, shown in Table 3.6.

Table 3.5: Activity Overlap on a subset of activities

| Activity | Stand | Walk | Left-Step | Left-Step Up |
|---|---|---|---|---|
| Stand | 1. | 0.138 | 0.313 | 0.136 |
| Walk | 0.138 | 1. | 0.333 | 0.119 |
| Left-Step | 0.313 | 0.333 | 1. | 0.114 |
| Left-Step Up | 0.136 | 0.069 | 0.114 | 1. |

Table 3.6: Activity confusion prediction

| Activity | Stand | Walk | Left-Step | Left-Step Up |
|---|---|---|---|---|
| Stand | 62.989 | 8.68 | 19.70 | 8.59 |
| Walk | 8.68 | 64.954 | 21.63 | 4.46 |
| Left-Step | 17.8 | 18.92 | 56.81 | 6.42 |
| Left-Step Up | 9.94 | 8.71 | 8.30 | 73.03 |

As and example using the Table 3.6, the predicted classification rate of the **Stand** activity is 62% overall, and it has a lower probability of being misclassified to the other activities as well. It is important to understand that misclassification is a directed relation, $A1 \rightarrow A2 \neq A2 \rightarrow A1$ where $\rightarrow$ indicates the direction of classification. The non-symmetrical relationship is due to the weighted distances of the activities.

Figure 3.20 provides visualization of where these overlaps occur. The examination in this section is restricted to the activities **Stand** and **Walk**. The standing activity (green) is plotted with the walking

(red). The circles represent the weights on the activity limbs and illustrates how an activity may envelope another within the activity space.

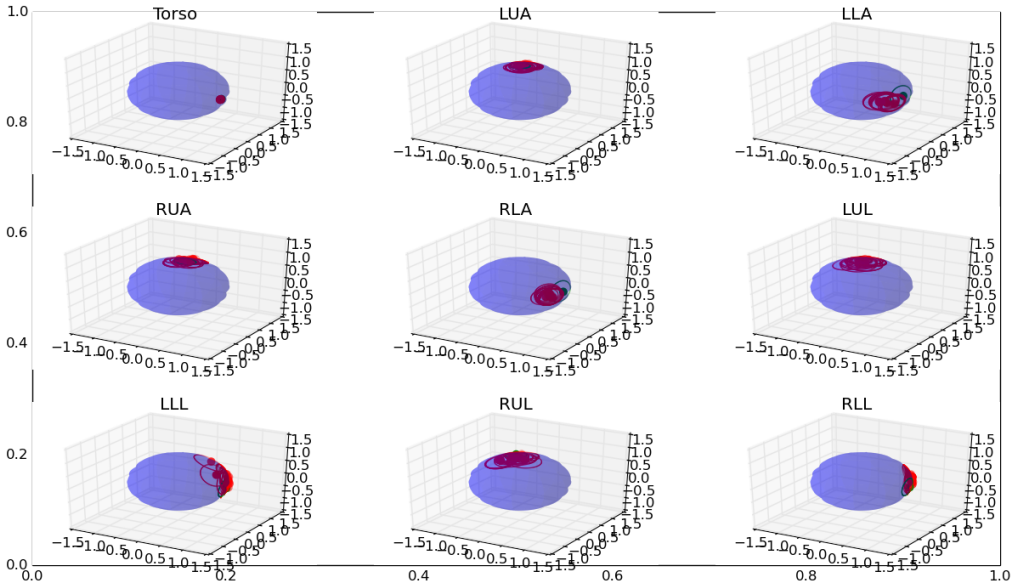



Figure 3.20: Example of Standing (green) and Walk (red) overlap

### 3.3.2 Error Function Adjustment

The errors of the body-worn system have been experimentally quantified in Section 3.2 to show that the system is noisier than a visual system. Therefore, even with filtering and noise mitigation calibration (online and offline) the classification of an activity is still challenging. As such, it is necessary to adjust the error function used to calculate the match of the activity. The error function given in 3.3,

$$\eta\left(\mathbf{x},\mathbf{y}\right)_{\mathbf{w}} = MAX\left(w_i \odot d_i\left(\mathbf{x},\mathbf{y}\right)\right) \tag{3.3}$$

where $w_i$ is element $i$ of $\mathbf{w}$, a vector of weights for each dimension of the activity. The error metric computes the weighted distance between two poses $p_1$ and $p_2$ and will return an error less than 1 if the pose matches an activity within the learned bounds. A threshold value of 1.77 was chosen in Blake (2014) for the activity rejection threshold. However, with a noisier system this is difficult to achieve, and it is necessary to consider scaling the threshold to account for the noise in the system. The average ratio of the archetype weights

is used to set a scaled threshold factor. The Motion Capture activity **Stand** and the Jumpsuit activity **Stand** were used to compute the scaling factor of 2.5 applied to a threshold value of 1. The ratio of the two systems allows a simple method to scale the threshold and prevent activities in the body-worn system from being erroneously classified as unknown. It is expected that the scaling of the threshold in this manner will introduce misclassification of unknown activities into the known set. The effect of scaling the threshold is examined in Chapter 5.

## 3.4 Framework

The methods examined in section 3.2 illustrate that there are a few key points to consider in transitioning a different sensor system to the quaternion space. A substantial portion of the effort is applied to understanding the underlying noise and error contributions that make up the system. Additionally, providing an error function to account for the issues that arise in a body worn system where error can propagate through a kinematically linked chain is also important. Stepping back from the sensor domain, this section examines the overarching goals of the framework and the necessary considerations that drive the successful transfer to the quaternion domain.

### 3.4.1 Noise and Error Consideration

As section 3.3.2 made clear, a simple combination of the underlying sensor measurements may contribute to noise. Therefore, it is essential to consider the sub-component error, mitigate at the low level, and then map to the higher level quaternion representation.

Additionally, an online calibration routine should be supplied to handle environmental changes or shifts in calibration in order to re-adjust bias within sensor devices. Specifically considered here are the gyrosope sensors. Once the bias shifts are accounted for the necessary pose level shifts should be corrected.

This study uses the sitting and standing poses to correct the body-level shift of the sensor placement. However, as was described in Section 3.2.9, it is necessary to understand the limits of the underlying system the pose is built upon. Appropriate measures should be taken to choose a pose that is sufficiently accurate to detect the offset errors and outside a highly error prone pose.

### 3.4.2 Placement and Fit

The garment dictates the wearability of the system. Placement of the sensors on a garment should be iteratively determined to facilitate ease of movement, mitigate fabric folding and bunching effects, and provide accurate detection of the activities. As will be discussed in Chapter 5, the fit of the garment plays a considerable role in the overall accuracy of the system. The first iteration of the jumpsuit achieved only a 38% overall accuracy in the activity classification while a refined version achieved a substantial improvement of approximately 51%. The sensors are the same for both jumpsuits and only the garment design changed. Consequently the improvements in the classifier accuracy are due, in part, to the change in garment.

# Chapter 4

# Prototype Development & User Study

This section details the prototype development of the Jumpsuit V2. Additionally, this section summarizes the functionality of the current suit and provides the distinctions where these changes deviate from the previous iterations. The user study is presented in this chapter and outlines the activity space, and discusses the overlap between certain activities.

## 4.1 Fit

The Jumpsuit garment is not form-fitting; however, there are bounds to how loose the garment may be. Given the weight of the sensors, the garment should not shift in static poses. Drape, defined as the extent to which a fabric will deform when it is allowed to hang under its own weight, is one of the most important fabric properties and plays an important role in garment appearance (Gioberto and Dunne, 2012). Different fabrics have different drape coefficients. While this doesn't specifically target the wearable electronics attached to a fabric, one can use it as a heuristic element to evaluate suitable fabrics. Similarly, movement during activities should not be completely masked by the fabric. Movement masking comes in two forms. If the fabric is exceedingly loose-fitting, then bending of the joints may be completely masked within the garment. Alternatively, movement artifacts can be created if the underlying fabric is too stiff or rolled up to fit the subject.

The one-size-fit-all mentality of the Jumpsuit V1 garment was woefully inadequate during the first user study. For each user, the jumpsuit was altered at the start of each test. The sensor location was adjusted by moving the communication ribbons up or down to adapt to the different arm lengths, with the extra

fabric in the arms taken-in by lacing created bunch points during bend movements. The extra fabric and lacing restricted movement during the operation of the user studies. These issues were mitigated as much as possible during the studies. The fit of the pants was generally too big for most of the users in the study. Even with the lace points in the pants and additional pins to tighten the pants up in key locations, this caused loss of fidelity of the measurements in the stepping and walking activities. The sensors were also able to shift location too freely during movements.

The second iteration of the jumpsuit was designed with these constraints under consideration. The commercial-of-the-shelf (COTS) jumpsuits were sized as small, medium and large to help accommodate a larger portion of the population and prevent the extra fabric bunching artifacts in the sensor measurements. An additional modification was made so that the wiring was not woven or sewn into ribbons to then connect to the sensor boards, but flat usb cables were used in channels sewn into the garments. The removal of the strain relief boards provided extra flexibility in placement of the sensors to help account for arm length variation in the users. This was further facilitated by the sensor mounting cases that snapped into the garment in 3 different straight-line locations to allow quick shifting of the sensors. The choice to move to the flat usb cables mitigated the connectivity effects of the IDC connectors. While this does provide an alternative to the IDC connectors, it is not ideal because the integration into the fabric substrate is more of an afterthought.

## 4.2 Electronics & Placement

The sensor boards and functionality were also extended from the previous iterations. The underlying communication on the I2C bus is unchanged. However, the communication to the Tier 2 node that combines the data from the Tier 1 sensors is updated to a Beagle Bone development board with a I2C to UART cape that provides the interface to the multi-master synced I2C communications and the Angstrom Linux kernel running on the Beagle Bone.

Focusing on the fusion of the data on the Tier 2 level, there are a few changes to help facilitate experimentation of the system and different sensors. The communication off board has been expanded by the Beagle Bone by allowing Ethernet, USB, or wireless capabilities through an SSH connection. This is in contrast to the previous RS-232 or Bluetooth capabilities that existed on the GUMSTIX board.

To account for different sensor variation and bias, the system has been augmented with an online calibration routine, which runs during the first 2 seconds of power-up of the motioncapture code base. It

assumes the user is standing still and zeros out the gyros. The data from the gyros and compass is combined to form the quaternion representation of the body model as described in Chapter 2.3. This provides the torso-referenced body model which can then be passed directly to the classifier.

The data structure differs from the structure outlined in Lewis (2011) thesis(4.1). The structure inserts a sensor identification flag **SensorID** to annotate which sensors contributed to the quaternion calculation, which is necessary to examine the effects of the different sensors on the overall accuracy and tracking.

**0xFE** and **0xFF** frame the data packet to enable easy separation of the data from each sensor. **QWU** and **QWL** represent the upper and lower byte of the first index of the four-element quaternion value. The value is the unsigned character representation of the float value. As the quaternion value is a four-element feature, the **Q*U** and **Q*L** represent the subsequent values of the quaternion. The **A*U** and **A*L** similarly represent the upper and lower bytes of the accelerometer value for the limb segment in consideration.

Table 4.1: Data structure of the resulting "packet" from the Tier 2 node, packet is split in half to accommodate the page

| 0xFE | LimbID | SensorID | QWU | QWL | QXU | QXL | QYU | QYL |
|---|---|---|---|---|---|---|---|---|
| QZU | QZL | AXU | AXL | AYU | AYL | AZU | AZL | OXFF |

## 4.3 Placement Considerations

The placement of the sensors was also considered during the re-design of the jumpsuit. The original strain relief boards added 2 inches to the overall length of the boards, which limited movement of the sensors to areas on the suit where joints and fabric bunching would not interfere. The movement away from the strain relief boards to the flat usb cabling as shown in Figure 4.1 provides more flexibility in placement. Additionally, to mitigate the overall sensor movement on the fabric substrate, additional snaps were added to allow 3 different placements along the limb segments. This allowed for quicker re-adjustment of the sensors on the subject to account for variations in segment length.

Figure 4.1: Snaps enable adjustment in a linear fashion along the primary axis of the limb segment

## 4.4 User Study

Two users studies were conducted with the body-worn garment over the course of a year. The users selected ranged in age from 20 to 38, with males making up 75% of the users. There is also a 50% overlap in users between the studies. This overlap allows a fit comparison between the two garments. Each user of the study participated in the tasks to the best of their ability. If the user was unable to accomplish the activity then it was skipped.

### 4.4.1 Selection of Activities and Specifics

The activities were selected to encompass a range of common physical therapy and everyday activities as well as to examine how well the classifier operated on highly disjoint activities and closely related activities. The selected activities are described in Table 4.2. There are 6 sitting and 13 standing activities. The user was asked to perform activities with their dominate hand, and the sequence was mirrored to the appropriate side.

| | |
|---|---|
| ***Stand*** | User stands in a neutral position with their feet forward and arms by their side. |
| ***Standing Arm Circles*** | User stands in a neutral position with their feet forward performs arm circles moving toward the front of the user and then rotating towards the back. These are performed at approximately a 45° angle from the shoulder joint. The users are asked to perform 3 cycles of the activity. angular rotation speed is approximately 1 Hz, or 1 rotation per second. |
| ***Standing Vertical Arm Circles*** | User stands in a neutral position with their feet forward performs vertical arm circles moving toward the front of the user and then rotating towards the back. These are performed at approximately a 90° angle from the shoulder joint. The users are asked to perform 3 cycles of the activity. The angular rotation speed is approximately 1 Hz, or 1 rotation a second. |
| ***Left Step*** | The user takes a step forward beginning with their left foot and completes the action by bringing the right foot next to their left. This is a normal step as one would take while walking. |
| ***Left Step Up*** | User takes a step forward / up in a natural manner with the left foot to the stool. They complete the motion by bringing the right foot next to the left foot on the stool. |
| ***Sit*** | User is asked to sit in a chair with their arms on the arm rests. Feet should be in a natural position, with the base of the foot on the floor. |
| ***Sitting Horizontal Arm Circles*** | Same method for standing arm circles, except the user is in a sitting position. |
| ***Sitting Vertical Arm Circles*** | Same method for standing vertical arm circles, except the user is in a sitting position. |
| ***Driving*** | User sits with feet out at pedals and hands at 10 and 2 on a steering wheel and pretends to drive. This is an automatic car and the user is driving on a straight road with little steering wheel correction. |
| ***Right Eat*** | User sits and eats with right hand as one would eat soup. Left hand is placed on the table. Feet are on the floor. Legs are not crossed. |
| ***Computer Use*** | User pretends to sit and type on a computer. The movement is typing at a keyboard for 1 second, followed by moving a mouse for 1 second and then return to the keyboard typing. |
| ***Right Shoulder Rotation*** | User is in a neutral standing position. Upper right arm pinned at side, hand on stomach, rotate hand and forearm out towards their right side to roughly 90° or comfortable rotation and back. The action is performed in approximately 3 seconds. |
| ***Button Shirt*** | User buttons a dress shirt with an imaginary three buttons. The movement begins at the top of the shirt, and moves down buttoning three buttons, top, middle, and bottom of the shirt. The duration of the activity is approximately 3 seconds. |
| ***Right Brush Hair*** | The user is to brush their hair with their right hand holding comb. The activity is restricted in domain to brushing a short cut hair forward. The upper right arm is positioned parallel to the floor. |
| ***Right Brush Teeth*** | The user pretends to brush their teeth in a cyclic manner. This motion is restricted to have the upper right arm parallel to the floor. The forearm movement is an elliptical movement. |
| ***T-Body Rotation*** | The activity is a cyclic activity. The user is requested to perform T-Body rotations. The movement begins with the users arms out in a T position. The rotation is a twisting motion of the torso. This movement should completely overlap a static T position. |
| ***Right Shoulder Rotation 2*** | The user positions the upper right arm parallel to floor in line their shoulders, forearm perpendicular to the upper arm in line with shoulders. The user rotates shoulder 90° forward keeping upper arm parallel to floor and inline with the shoulders, and bringing the forearm parallel to the floor facing to the front of the torso and returns back to start. The motion is to take approximately 2-3 seconds. |
| ***Bicep curls*** | The user is asked to pin their elbows at their side and perform bicep curls. The forearms are to be shoulder width apart and maintain the same speed and rotational position as each other. This is facilitated by the use of a PVC pipe to act as a curl bar. |
| ***Walk*** | The user is asked to walk forward in a straight line path for approximately 6 steps while maintaining normal posture and gait. |

Table 4.2: User study activities

# Chapter 5

# Results and Discussion

This chapter presents the results from the user study described in Chapter 4, highlights the overall performance of the prototype, and examines contributing factors that determine measurement quality. Sensor independence can be shown by comparing the body-worn system to the archetypes of the visual systems. Investigation into misclassified activities is presented using activity overlap to facilitate an understanding as to why one activity can be confused with another. A summary of the overall results is given in Section 5.3.

## 5.1 Prediction using Activity Overlap

The overlap method outlined in Section 3.3.1 is used to examine the underlying probability of misclassification and is computed on the original Kinect data set from Blake (2014). This enables the evaluation of misclassification modes without the added error introduced by different sensor systems. Table 5.1 presents the original classification of the dataset, Table 5.2 presents the predicted classification of the dataset activities, and Table 5.3 provides a quick reference to the difference between the two tables. The overall accuracy of the predicted versus the actual is within 4% of each other. The misclassification error of an activity can be due to three different components: measurement error, activity similarity, and alphabet substring matching.

***Activity Similarity:*** The overlap of an activity is computed from the archetype. As such, if the observed activity set was performed sufficiently many times, then it should approach the classification accuracy. The prediction does not consider peaked distributions of the activity space; therefore, the overlap method may not capture variations within the activity set that skew an activity distribution.

***Measurement Error:*** It is inherently assumed that the archetypes capture the full variation of an

activity. The variation of an activity is assumed to be greater than the measurement error of the system being examined. Therefore, if there is measurement error within the system that biases the activity distribution within an overlapped region, the predication may fail to capture this result.

***Alphabet Substrings:*** Alphabet substring matching occurs when there is similarity within the activity space between two activities. The substring match is generally specific to activities where one has a shorter duration than the confused activity. Substring matching is an issue as activities have valid sub-activities that coexist within the larger activity. If the activity alphabets were forced to be the same length, then the substring matching would be mitigated. A simplified example of how this can occur is shown in Figure 5.1, which illustrates activity **A2** matching with a lower error, E=0, to the observed sequence. The activities are **A1** and **A2** and are comprised of characters 1 through 6.

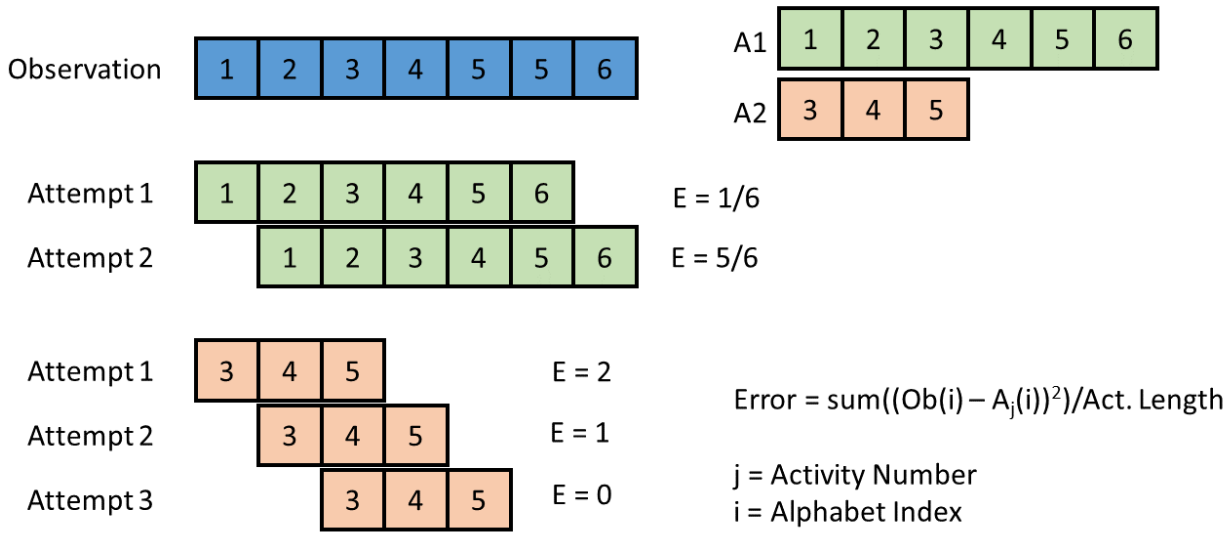



Figure 5.1: Example substring match and how it can occur in the system

### Activity Prediction Comparison

Examining **Left Step** and **Left Step Up** in Table 5.1 illustrates that the predicted confusion in Table 5.2, by comparison, is overly pessimistic. This discrepancy is explained by considering that the distinguishing limbs between the two activities are the upper legs and their angular movement. The separation of the activities is due to the additional upward movement that prevents the **Left Step** activity from matching to

**Left Step Up**.

Examining **Right Brush Hair** and **Right Brush Teeth**, the predicted accuracy of the **Right Brush Hair** example is within 5% of the classified observed activities. The misclassification discrepancy between the predicted and actual **Right Brush Teeth** is most likely due to a peaked error distribution towards **Right Brush Hair** activity; this discrepancy is not accounted for in the overlap metric.

The prediction results are encouraging for the Kinect, which has lower measurement error than the Jumpsuit. This distinction is important, as the overlap method is computed on the archetypes and provides an idealized prediction. As system error increases, the misclassification will increase as the threshold is expanded to account for the increase in error. The increase in the threshold has the effect of allowing poorer representations of the activity to match, which may include activities that are not the correct activity.

| | LeftStep | LeftStepUp | ShoulderCircles | Rt.ShoulderRot | RightBrushTeeth | RightBrushHair | Driving | RightEat | Unkown |
|---|---|---|---|---|---|---|---|---|---|
| LeftStep | **92.0** | 6.7 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| LeftStepUp | 0.00 | **94.6** | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 5.40 |
| ShoulderCircles | 0.00 | 0.00 | **96.0** | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 4.00 |
| RightShoulderRotation | 0.00 | 0.00 | 1.4 | **95.8** | 0.00 | 0.00 | 0.00 | 0.00 | 2.8 |
| RightBrushTeeth | 0.00 | 0.00 | 0.00 | 0.00 | **48.6** | 44.30 | 0.00 | 0.00 | 7.1 |
| RightBrushHair | 0.00 | 0.00 | 0.00 | 0.00 | 4.6 | **80.3** | 0.00 | 0.00 | 15.1 |
| Driving | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | **80.0** | 17.3 | 2.7 |
| RightEat | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 7.4 | **89.7** | 2.9 |

Table 5.1: Kinect activities classified to the Kinect archetypes. Overall accuracy 84.6%.
**Source:** Blake (2014)

| | LeftStep | LeftStepUp | ShoulderCircles | Rt.ShoulderRot | RightBrushTeeth | RightBrushHair | Driving | RightEat |
|---|---|---|---|---|---|---|---|---|
| LeftStep | **82.607** | 17.111 | 0. | 0. | 0. | 0. | 0. | 0. |
| LeftStep Up | 17.393 | **81.272** | 1.38 | 0.586 | 0. | 0. | 0. | 0. |
| Shoulder Circles | 0. | 1.137 | **98.62** | 0. | 0. | 0. | 0. | 0. |
| Right Shoulder Rotation | 0. | 0.479 | 0. | **99.414** | 0. | 0. | 0. | 0. |
| Right Brush Teeth | 0. | 0. | 0. | 0. | **75.627** | 24.373 | 0. | 0. |
| Right Brush Hair | 0. | 0. | 0. | 0. | 24.373 | **75.627** | 0. | 0. |
| Driving | 0. | 0. | 0. | 0. | 0. | 0. | **98.39** | 1.61 |
| Right Eat | 0. | 0. | 0. | 0. | 0. | 0. | 1.61 | **98.39** |

Table 5.2: Activity classification prediction using the overlap method for the Kinect System. Overall predicted accuracy 89.0%

| | LeftStep | LeftStepUp | ShoulderCircles | Rt.ShoulderRot | RightBrushTeeth | RightBrushHair | Driving | RightEat |
|---|---|---|---|---|---|---|---|---|
| Left Step | **9.4** | -10.41 | 0 | 0 | 0 | 0 | 0 | 0 |
| Left Step Up | -17.393 | **12.33** | -1.38 | 0 | 0 | 0 | 0 | 0 |
| Shoulder Circles | 0 | -1.137 | **-2.62** | -0.586 | 0 | 0 | 0 | 0 |
| Rt. Shoulder Rotation | 0 | -0.479 | 1.4 | **-3.61** | 0 | 0 | 0 | 0 |
| Right Brush Teeth | 0 | 0 | 0 | 0 | **-27.02** | 19.93 | 0 | 0 |
| Right Brush Hair | 0 | 0 | 0 | 0 | -19.77 | **4.68** | 0 | 0 |
| Driving | 0 | 0 | 0 | 0 | 0 | 0 | **-18.39** | 15.99 |
| Right Eat | 0 | 0 | 0 | 0 | 0 | 0 | 5.79 | **-8.69** |

Table 5.3: Difference between the predicted and actual classification due to the inter-class confusion from Table 5.1 and Table 5.2

## 5.2 Validating Sensor Independence

To validate the sensor independence of the classification system, the Jumpsuit is compared to both the Kinect archetypes and the Motion Capture archetypes. In order to examine the systems in a side-by-side manner the various systems are mapped to the nine-dimensioned body model, as was described in Section 2.4. The mapping considers the joint-locations from the vision based systems and constructs the limb positions in quaternion sequence form. The comparison to the two systems enables a closer examination into the misclassification modes of the body-worn system to a smaller activity set and a larger activity set. The discussions that follow examine specific examples within each comparison to understand the misclassified examples and to give a more complete understanding of the errors encountered with an expanded activity set.

### 5.2.1 Jumpsuit vs. Kinect

| | LeftStep | LeftStepUp | ShoulderCircles | Rt.ShoulderRot | RightBrushTeeth | RightBrushHair | Driving | RightEat | Unkown |
|---|---|---|---|---|---|---|---|---|---|
| LeftStep | **66.67** | 33.33 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| LeftStepUp | 68.89 | **31.11** | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| ShoulderCircles | 0.00 | 31.11 | **62.22** | 2.22 | 0.00 | 4.44 | 0.00 | 0.00 | 0.00 |
| RightShoulderRotation | 17.78 | 2.22 | 0.00 | **80.00** | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| RightBrushTeeth | 0.00 | 0.00 | 0.00 | 0.00 | **82.22** | 17.78 | 0.00 | 0.00 | 0.00 |
| RightBrushHair | 0.00 | 0.00 | 0.00 | 0.00 | 35.56 | **64.44** | 0.00 | 0.00 | 0.00 |
| Driving | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | **17.78** | 64.44 | 17.78 |
| RightEat | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 2.22 | **86.67** | 11.11 |

Table 5.4: Jumpsuit activities classified to the Kinect archetypes. Overall accuracy 61.4%

Using the Kinect archetypes, the Jumpsuit has been compared directly to the original system in Blake (2014). The misclassification error of the Jumpsuit is examined with respect to the *Activity Similarity*, *Mea-*

*surement Error*, and *Alphabet Substrings*. The results are generated with an error threshold for classification set to 2.4.

Examining Table 5.4, the confusion for activities **Left Step** and **Left Step Up** arise due to the lower fidelity of the body-worn system. As such, the distance between the angular movement of the upper legs, which distinguishes these activities, is unable to be detected within the body-worn system. This illustrates a key aspect to consider for the viability of a system for a particular application. The lower fidelity is due to a combination of the sensor error and the loose fitting nature of the garment, described in Section 3.2.

An example of substring matching that is possible as the system error increases is the **Shoulder Circles** activity matching to **Left Step Up** activity. In a similar manner to the example given in Figure 5.1, the quaternions that represent the arms at the low point (arms down near the subjects side) of the shoulder circle activity match to the **Left Step** and **Left Step Up** activity. The final classification to the **Left Step Up** activity is due to the lower weights on the legs, which generate a lower error match compared to the **Left Step** activity.

The **Right Shoulder Rotation** misclassification to the **Left Step** activity represents the issue of expanding the error threshold to account for the measurement error. These activities are only classified to the **Left Step** because they match below the threshold for the activity. However, in examining the computed error on the actual activity of interest (Right Shoulder Rotation) they would normally be classified as incorrect.

The **Driving** misclassification to the **Right Eat** activity is due to a combination of measurement error and the lower weights on the **Right Eat** activity. This causes the **Driving** activity to match with a lower score to **Right Eat**. The measurement error with the Jumpsuit creates motion in the arms that becomes misinterpreted as an eating motion.

### 5.2.2 Jumpsuit vs. Motion Capture

In Table 5.5 the Jumpsuit is compared to the Motion Capture system. The overall classification is reduced from the Kinect system due to higher activity overlap in certain activities, and test-subject deviation from the prescribed activity.

The analysis in this section considers the misclassification among the sitting activities, which are broken out in Table 5.7. The activity space within the sitting activities is predicted to be highly confused (Table 5.6). This presents an additional challenge when compared to the previous Kinect set, where the activities were not as highly confused. **Sit Computer Usage** encompasses a large portion of **Sit Chair**, **Driving**, and **Right**

| | 1: Stand | 2: LeftStep | 3: LeftStepUp | 4: Walk | 5: ShoulderCircles | 6: Rt.ShoulderRot | 7: RightBrushTeeth | 8: RightBrushHair | 9: Stand Button Shirt | 10: StandVrt. ShoulderCircles | 11: TBodyRotation | 12: Right Shoulder Rot. 2 | 13: Bicep Curls | 14: SitChair | 15: Driving | 16: RightEat | 17: Sit Computer Usage | 18: Sit Vrt. Shoulder Circles | 19: Sit Hor. Shoulder Circles | 20: Unkown |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | **100.00** | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 2 | 33.33 | **28.89** | 0.00 | 11.11 | 0.00 | 26.67 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 3 | 15.56 | 46.67 | **0.00** | 20.00 | 0.00 | 17.78 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 4 | 31.11 | 13.33 | 0.00 | **13.33** | 0.00 | 42.22 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 5 | 0.00 | 0.00 | 0.00 | 0.00 | **68.89** | 0.00 | 2.22 | 0.00 | 0.00 | 4.44 | 24.44 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 6 | 0.00 | 0.00 | 0.00 | 2.22 | 0.00 | **97.78** | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 7 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | **84.44** | 13.33 | 0.00 | 0.00 | 0.00 | 2.22 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 8 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 11.11 | **88.89** | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 9 | 4.44 | 17.78 | 0.00 | 4.44 | 0.00 | 22.22 | 0.00 | 0.00 | **40.00** | 0.00 | 0.00 | 0.00 | 11.11 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 10 | 0.00 | 0.00 | 0.00 | 0.00 | 2.86 | 0.00 | 0.00 | 0.00 | 0.00 | **97.14** | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 11 | 0.00 | 2.22 | 0.00 | 2.22 | 28.89 | 0.00 | 4.44 | 0.00 | 0.00 | 0.00 | **62.22** | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 12 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 11.11 | 0.00 | 0.00 | 0.00 | 0.00 | **88.89** | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 13 | 37.21 | 37.21 | 0.00 | 11.63 | 0.00 | 4.65 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | **9.30** | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| 14 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | **0.00** | 4.44 | 0.00 | 95.56 | 0.00 | 0.00 | 0.00 |
| 15 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | **40.00** | 0.00 | 51.11 | 8.89 | 0.00 | 0.00 |
| 16 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | **0.00** | 100.00 | 0.00 | 0.00 | 0.00 |
| 17 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 4.44 | 4.44 | 0.00 | **91.11** | 0.00 | 0.00 | 0.00 |
| 18 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 10.00 | **32.00** | 38.00 | 20.00 |
| 19 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 3.64 | 10.91 | 0.00 | 9.09 | 0.00 | **67.27** | 9.09 |

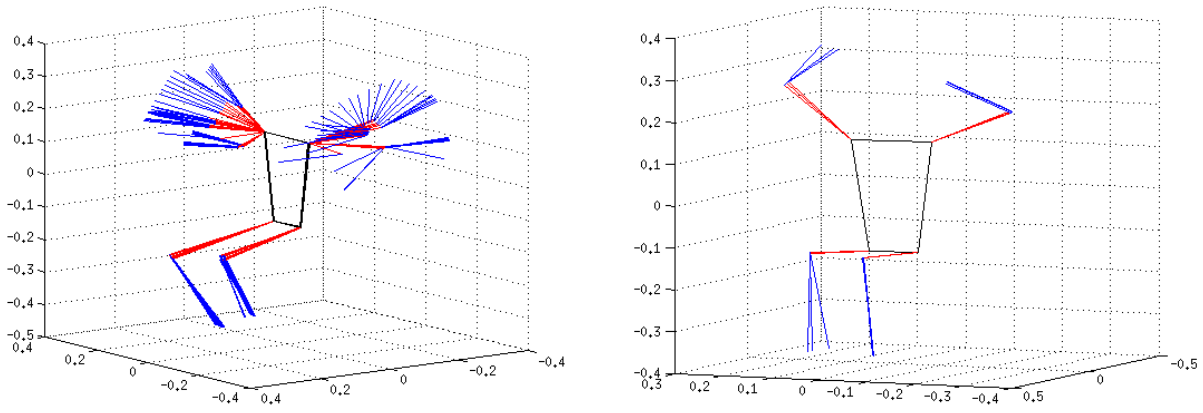
Table 5.5: Jumpsuit activities classified to the MoCap archetypes. Overall 53.4%

**Eat**, shown in Table 5.6. Additionally, since the weights of the **Sit Computer Usage** activity are low by comparison to the other activities, it is an activity containing high variability in allowable performance. The low weights of this activity allow overlapping activities to match with a lower score and be classified to the lower weight activity. The low score matching is particularly problematic because **Sit Chair** and **Right Eat** are fully enveloped by **Sit Computer Usage**.

The deviation from the prescribed activity **Sit Vertical Shoulder Circles** will be examined to explain the confusion with **Sit Horizontal Shoulder Circles**. The structure of these activities is designed to avoid confusion with each other; however, confusion arises due to incorrect execution of the activity. This is illustrated in Figure 5.2a, where the user in the vertical shoulder circle activity drops their upper arms to the incorrect position, allowing a fake match. This is compared to a more correct completion of the activity in Figure 5.2b.

| | Sit Chair | Driving | Right Eat | Sit Comp. Usage | Sit Vrt. Shoulder Circles | Sit Hor. Shoulder Circles |
|---|---|---|---|---|---|---|
| Sit Chair | 62.60 | 12.00 | 0.90 | 20.23 | 0.00 | 0.00 |
| Driving | 12.00 | 46.04 | 25.84 | 18.06 | 0.00 | 10.55 |
| Right Eat | 1.00 | 19.24 | 61.82 | 10.12 | 0.00 | 2.209 |
| Sit Comp. Usage | 24.53 | 16.09 | 9.57 | 51.62 | 0.00 | 0.00 |
| Sit Vrt. Shoulder Circles | 0.00 | 0.00 | 0.00 | 0.00 | 83.78 | 14.15 |
| Sit Hor. Shoulder Circles | 0.00 | 6.65 | 1.87 | 0.00 | 16.22 | 73.09 |

Table 5.6: Confusion prediction between sitting activities from the Qualisys motion capture system



(a) Example of a bad sitting vertical shoulder circle activity (b) Example of a correct vertical shoulder circle activity

Figure 5.2: Visual comparison of the vertical shoulder circle activities from the Jumpsuit

| | SitChair | Driving | Right Eat | Sit Computer Usage | Sit Vir. Shoulder Circles | Sit Hor. Shoulder Circles | Unkown | Percent |
|---|---|---|---|---|---|---|---|---|
| Sit Chair | **0.00** | 4.44 | 0.00 | 95.56 | 0.00 | 0.00 | 0.00 | |
| Driving | 0.00 | **40.00** | 0.00 | 51.11 | 8.89 | 0.00 | 0.00 | |
| Right Eat | 0.00 | 0.00 | **0.00** | 100.00 | 0.00 | 0.00 | 0.00 | |
| Sit Computer Usage | 4.44 | 4.44 | 0.00 | **91.11** | 0.00 | 0.00 | 0.00 | |
| Sit Vrt. Shoulder Circles | 0.00 | 0.00 | 0.00 | 10.00 | **32.00** | 38.00 | 20.00 | |
| Sit Hor. Shoulder Circles | 3.64 | 10.91 | 0.00 | 9.09 | 0.00 | **67.27** | 9.09 | |

Table 5.7: Jumpsuit sitting activities

## 5.3 Summary of Results and Discussion

In comparing the Jumpsuit to the Kinect and Motion Capture activity archetypes, it was shown that the body-worn system was able to classify to the vision-based archetypes. However, due to *Activity Similarity*, *Measurement Error*, and *Alphabet Substrings*, misclassification of the observed activities resulted. The overall results are presented in Table 5.8. Jumpsuit V1 refers to the first version of the body-worn system. Jumpsuit V2 refers to the current system referenced in Section 2.

The results in Table 5.8 indicate that the one-size-fit-all garment construction that was used in V1 was not sufficient due to the low overall accuracy. However, the difference in accuracy is not strictly related to the garment construction and is a combination of the methods of bias calibration from Section 3.2.9. The results were computed with the minimum bounding SLERP driven filter as defined in Section 3.2.8.

| | Kinect | Qualisys MoCap |
|---|---|---|
| Jumpsuit V1 | 32.20 | 28.5 |
| Jumpsuit V2 | 61.40 | 53.40 |

Table 5.8: Overall classification accuracy of the Jumpsuit activities to the vision-based archetypes

### 5.3.1 Threshold Implications

It is important to understand the implications of setting the threshold for the classifier and its relation to the activity space. The threshold defines what is considered a good match. As such, setting the threshold too high allows poor matches or activities that are not in the known set to match. This reduces the precision of the classification. Therefore, $F_1$ scores are used to further enforce the selection of the threshold at 2.2 for the body-worn system. $F_1$ scores provide a method to evaluate multi-class classification accuracy. Micro $F_1$ scores evaluate precision / recall on a class by class basis. Macro $F_1$ scores evaluate the precision / recall as a multi-class average. As the threshold increases, the activities that are allowed to match to any known activity grow. This presents a point where one may trade precision for recall in the system. Additionally, as the threshold increases, the unknown set of activities becomes further enveloped by the activity space and can generate false positive matches. In order to examine the false positive matching effect, a random subsample of activities was selected. Figure 5.3 illustrates the comparison between the different subsampled sets of activities. The knee in the graph at approximately 2.2 represents the threshold setting that should maximize accuracy while minimizing precision and recall error. The knee reinforces the computed value of 2.4 arrived at in Section 3.2. The dashed line indicates the threshold setting at 2.2.

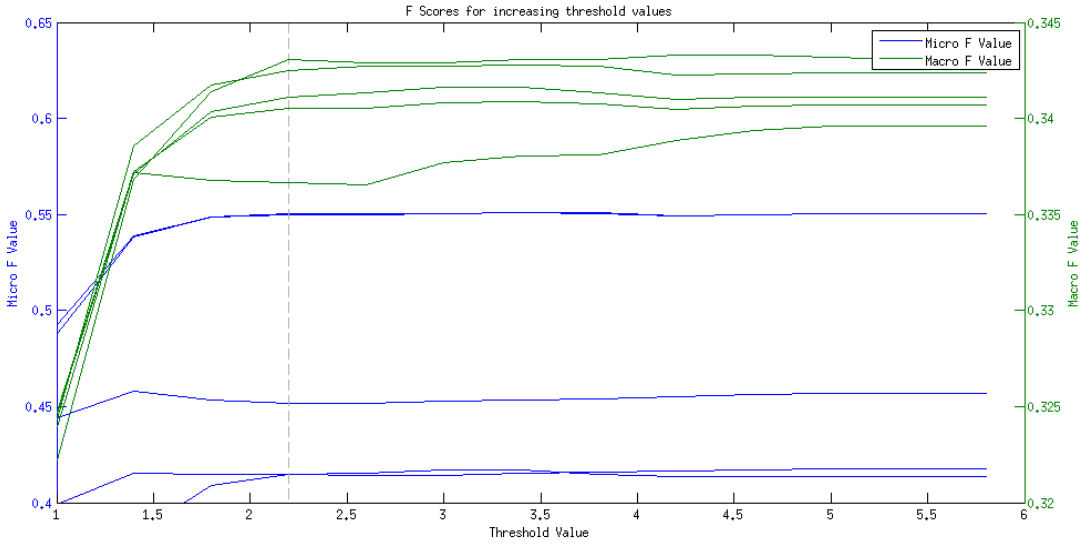



Figure 5.3: FScore (Macro / Micro) for the subsets of activities

### 5.3.2 Limitations from Overlap

Using the overlap method from Section 3.3.1, it is possible to examine how the predicted inter-class confusion changes as a function of the threshold. The analysis builds on the fact that the radii of the activities represent the bounds of a good activity. As the threshold expands, the boundary encompasses more activities that would normally be rejected as incorrect. Therefore, the experiment increases the threshold of the activities to provide an illustration showing where systems operate with respect to the ideal activities. This allows the visualization of the expected inter-class confusion based on the threshold of allowable match, shown in Figure 5.4.

It is important to understand what Figure 5.4 indicates. As the threshold grows, assuming there is a uniform distribution of activities within their respective activity spaces, the lower weights of an activity grow more quickly to encompass the space. A low weight indicates high variation and high weights indicate low variation for a given activity. The lower weights generate areas which overlap and eventually overcome all activities within the space, as such, the unknown or poorly matched activities will match to these "larger" activities.

An illustrative example is shown in Figure 5.5. Given an A, and B activity, the threshold of accepted activities is expanded. As the threshold expands, the envelope of acceptable matches to A and B expands.Given that B is defined to have a lower weight, the envelope of allowable matches will grow more quickly. Therefore, at a threshold, $T$, activity A will cease to be considered an appropriate match, and activity B will be the

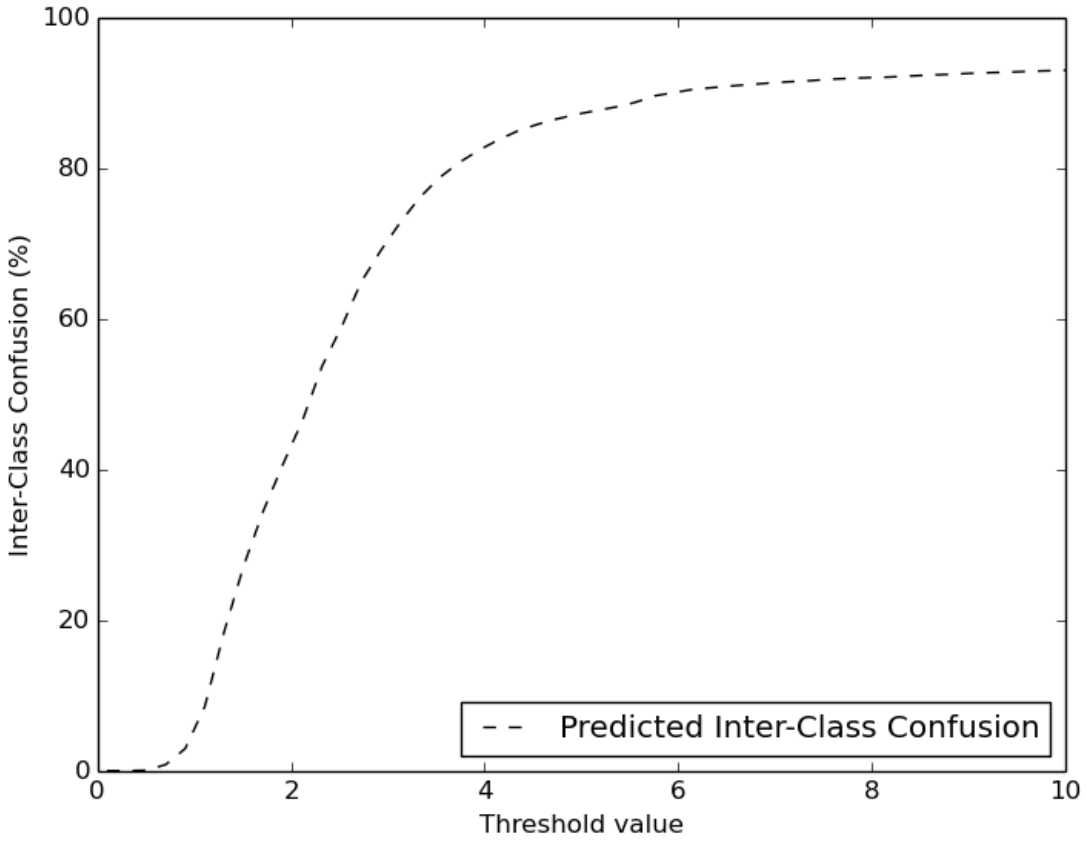



Figure 5.4: Illustrates the reduction of overall accuracy based on the inter-class confusion. The prediction is computed with the Motion Capture archetypes.

lower scoring match. The dashed lines indicate the activity space of A, where $R_A$ and $T_A$ show the original threshold and expanded threshold for activity A respectively. The shaded region indicates the area within activity space A that would be matched to A. The overlap effect within the activity space illustrates the trade-off between precision and recall of the system. Figure 5.6 shows the expected precision of the system at a given threshold setting. The plot shows that accounting for measurement errors within the system decreases the precision of activity classification. In order to examine the precision reduction within the Jumpsuit system, the observed activities for the body-worn system were plotted as a function of threshold increase.The plotted Jumpsuit data ends once the available observations are classified. The Jumpsuit results suggest that the predicted trend is a valid idealization. The observed precision of the system may fall on the right side of the predicted precision curve for a given set of observed activities, which indicates that the sampled population performed the activities in a peaked manner. The expectation is that as the sample

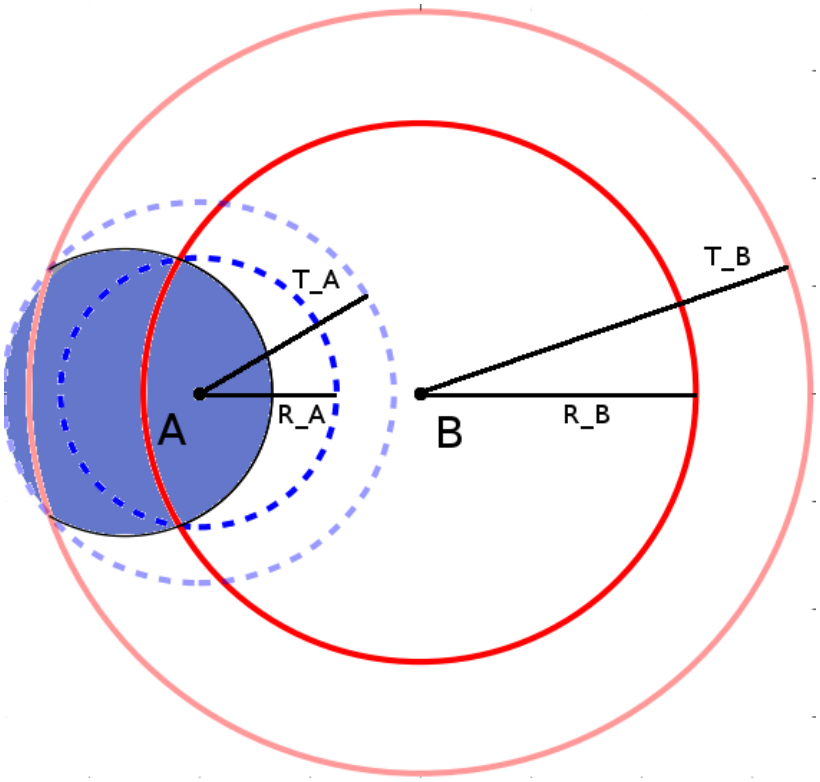



Figure 5.5: Example of threshold expansion. Blue shaded region indicating the area that would be correctly classified to activity A. $R$ and $T$ indicate the current and previous threshold values for the classes.

of the subjects expands, the distribution of allowable variations within the activity space would be more uniform and become a better match to the curve.

Section 5.2.1 and Section 5.2.2 showed that the majority of misclassification was due to the inter-class confusion for the given threshold of 2.2. Therefore, the Jumpsuit V2 system is within the expected overall accuracy range due to the inter-class confusion and would not be expected to increase overall accuracy much beyond the current value. It is understood however, that enforcing rigid placement of the sensors on the body would increase results as this would reduce the error compensation necessary. However, this would detract from the loose-fitting goal of the system.

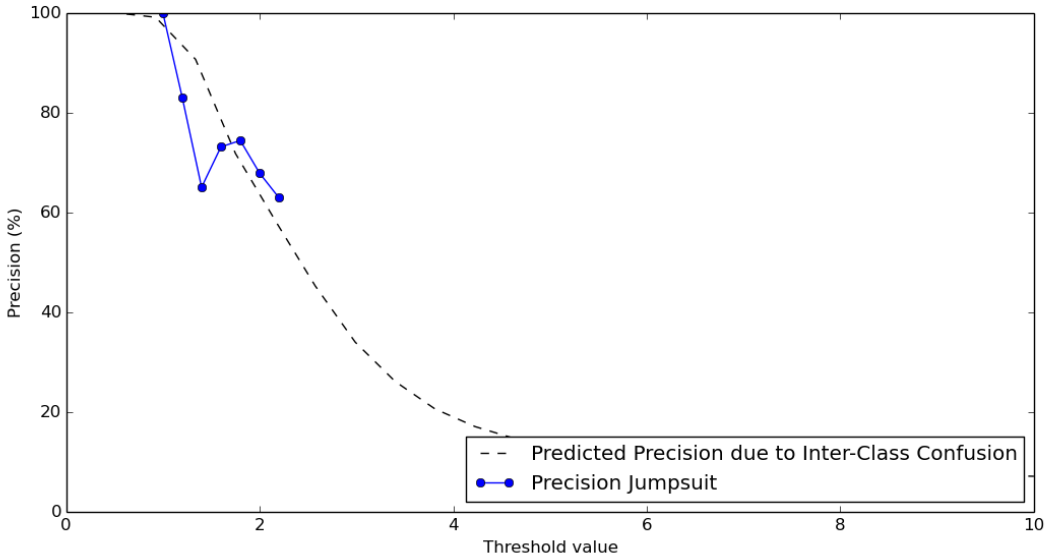



Figure 5.6: Precision trade-off as the threshold increases.

# Chapter 6

# Conclusion

## 6.1 Implications

This study has shown that the vision based classifier MUAC can be successfully adapted to a body-worn system. This is important, as it confirms that MUAC is indeed a sensor-independent classifier. The study illustrates the necessary considerations to transition the body-worn sensor-system to MUAC, and provides a framework to investigate the applicability of new sensors. These results support the use of MUAC as a sensor-independent activity classifier.

However, as activity overlap increases, the ability to distinguish between activities becomes impaired (Chapter 3). The reduction was also shown to be compounded by increases in measurement error. Therefore, it is necessary consider both the measurement error and the activity space when evaluating the applicability of a body-worn system. By reducing the evaluated activity space, overall accuracy can rise drastically, and increasing activity separation may provide a greater acceptance of the system.

## 6.2 Future Work

A great deal of attention within this study focused on the body-worn system and evaluation of classification challenges that it presented. Classification extension methods have been presented here that will increase overall accuracy, and enable greater separation of activities within the activity space. Of the most promising methods, expansion of the feature space would allow for greater separation of the activities. The ability to re-rank a classified activity based on a set of additional features such as geo-location or external cues would

enable a more complete separation of confused activities, a benefit that would only add a small computational component to the system.

A further study to examine how transitions between the activities occur would provide additional probability cues that the classifier system could integrate to improve classification. The transitions could be provided though an adaptation of Markov chains to incorporate a probabilistic transition between activities.

Investigation of a time-varying feature extraction method to allow direct comparison to fixed feature based classification methods such as k-nearest neighbour (kNN) or neural-networks may provide an alternate method to the current classification algorithm used. The two main drawbacks to a system built upon these methods is that the variation within activities is not necessarily captured in an intuitive manner, and they would necessitate a re-training of the system for each additional class. Therefore, unless the system was restricted to a set of known activities these methods would not allow the flexible addition of classes without re-training. An approach of this nature would limit the body-model flexibility across sensor domains and is not suggested as a beneficial extension.

As a final note, the classifier was centered around activity recognition of the human body and was constructed with a nine-dimensional feature vector. However, it can be extended to operate with more or less dimensions for a given body, which is noted in (Blake, 2014). For example, if one desired to detect only the movement of one limb, the system could be reduced to 2-dimensions of Upper and Lower limb movement. Conversely, if one had access to head movement as well the system could be expanded to incorporate head movement into the classification of an activity. Additionally, the concept can be applied to non-human bodies and therefore be of use in evaluating sequential action recognition of systems of linked bodies.

## 6.3 Final Remarks

As body-worn electronics become increasingly interwoven into our everyday lives, self-quantification may become the status quo, which would benefit from classification systems working across sensor domains. As such, this work provides an understanding into how a body-worn system transitions to a vision-based classifier and shows both where MUAC was successful and where issues can arise. With additional activity separation and smaller body-worn electronics, the body-worn system has the potential to become comparable to a vision based system.

# Bibliography

*Visualizing Quaternions.* Morgan Kaufmann, 2006.

O. Banos, A. Calatroni, M. Damas, H. Pomares, I. Rojas, H. Sagha, J. del R. Milln, G. Troster, R. Chavarriaga, and D. Roggen, "Kinect=IMU? Learning MIMO Signal Mappings to Automatically Translate Activity Recognition Systems across Sensor Modalities," *2012 16th International Symposium on Wearable Computers*, pp. 92–99, Jun. 2012. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6246149

M. T. Blake, "An Ambulatory Monitoring Algorithm to Unify Diverse E-Textile Garments," Master's thesis, Virginia Polytechnic and State University, 2014.

U. Blanke and B. Schiele, "Remember and transfer what you have learned-recognizing composite activities based on activity spotting," *Wearable Computers (ISWC), 2010 ...*, 2010. [Online]. Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5665869

K. Forster, D. Roggen, and G. Troster, "Unsupervised Classifier Self-Calibration through Repeated Context Occurences: Is there Robustness against Sensor Displacement to Gain?" *2009 International Symposium on Wearable Computers*, pp. 77–84, Sep. 2009. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5254652

F. Gemperle, C. Kasabach, J. Stivoric, M. Bauer, and R. Martin, "Design for wearability," *Digest of Papers. Second International Symposium on Wearable Computers (Cat. No.98EX215)*, pp. 116–122. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=729537

G. Gioberto and L. E. Dunne, "Garment Positioning and Drift in Garment-Integrated Wearable Sensing," *2012 16th International Symposium on Wearable Computers*, pp. 64–71, Jun. 2012. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6246144

Honeywell, "Three-axis Compass with Algorithms."

J. Lester, T. Choudhury, and G. Borriello, "A Practical Approach to Recognizing Physical Activities," pp. 1–16, 2006.

R. Lewis, "Analysis of a self-contained motion capture garment for e-textiles Analysis of a self-contained motion capture garment for e-textiles," 2011.

G. Navarro, "A guided tour to approximate string matching," *ACM Computing Surveys*, vol. 33, no. 1, pp. 31–88, Mar. 2001. [Online]. Available: http://portal.acm.org/citation.cfm?doid=375360.375365

G. Ogris, T. Stiefmeier, P. Lukowicz, and G. Tr, "Using a complex Multi-modal On-body Sensor System for Activity Spotting," *2008 12th IEEE International Symposium on Wearable ComputersInternational Symposium on Wearable Computers*, pp. 55–62, 2008.

A. Zinnen, U. Blanke, and B. Schiele, "An Analysis of Sensor-Oriented vs. Model-Based Activity Recognition," *2009 International Symposium on Wearable Computers*, pp. 93–100, Sep. 2009. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5254654

A. Zinnen, C. Wojek, and B. Schiele, "Multi Activity Recognition Based on Bodymodel-Derived Primitives," pp. 1–18, 2009.