University of Tennessee, Knoxville
TRACE: Tennessee Research and Creative Exchange

Doctoral Dissertations | Graduate School

12-2012

Quaternionic Attitude Estimation with Inertial Measuring Unit for Robotic and Human Body Motion Tracking using Sequential Monte Carlo Methods with Hyper-Dimensional Spherical Distributions

Gary To
gto@utk.edu

Follow this and additional works at: https://trace.tennessee.edu/utk_graddiss

Part of the Biomedical Devices and Instrumentation Commons

Recommended Citation

To, Gary, "Quaternionic Attitude Estimation with Inertial Measuring Unit for Robotic and Human Body Motion Tracking using Sequential Monte Carlo Methods with Hyper-Dimensional Spherical Distributions. " PhD diss., University of Tennessee, 2012.
https://trace.tennessee.edu/utk_graddiss/1592

This Dissertation is brought to you for free and open access by the Graduate School at TRACE: Tennessee Research and Creative Exchange. It has been accepted for inclusion in Doctoral Dissertations by an authorized administrator of TRACE: Tennessee Research and Creative Exchange. For more information, please contact trace@utk.edu.

To the Graduate Council:

I am submitting herewith a dissertation written by Gary To entitled "Quaternionic Attitude Estimation with Inertial Measuring Unit for Robotic and Human Body Motion Tracking using Sequential Monte Carlo Methods with Hyper-Dimensional Spherical Distributions." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Biomedical Engineering.

Mohamed Mahfouz, Major Professor

We have read this dissertation and recommend its acceptance:

Richard Komistek, William Hamel, Aly Fathy

Accepted for the Council:

Carolyn R. Hodges

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Quaternionic Attitude Estimation with Inertial Measuring Unit for Robotic and Human Body Motion Tracking using Sequential Monte Carlo Methods with Hyper-Dimensional Spherical Distributions

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Gary To

December 2012

Copyright by
Gary To
2012

"Out of the night that covers me,
Black as the Pit from pole to pole,
I thank whatever gods may be
For my unconquerable soul.

In the fell clutch of circumstance
I have not winced nor cried aloud.
Under the bludgeonings of chance
My head is bloody, but unbowed.

Beyond this place of wrath and tears
Looms but the Horror of the shade,
And yet the menace of the years
Finds, and shall find, me unafraid.

It matters not how strait the gate,
How charged with punishments the scroll,
I am the master of my fate:
I am the captain of my soul."

Invictus, William Ernest Henley

I dedicate this dissertation to my mother, Shirley Ho, for her continuous support, love, and teaching me the importance of dreaming. I also dedicate this to my sister, Carine To, who has bought much excitement and joy in my life.

Acknowledgement

I would like to thank my advisor and mentor Dr. Mohamed Mahfouz for giving me this opportunity to excel in medical research. During the first few years working with Dr. Mahfouz, he has provided me with a lot of guidance and critical thinking crucial to engineering designs. I'm also very grateful that he gives me the flexibility to determine solutions through multiple different avenues on my own, and one of which leads to the research presented here in this dissertation. Dr. Mahfouz is also a good friend which has provided me personal support in many occasions.

I would also like to extend my thanks to Dr. Komistek. He has given me opportunity to work with him in the Center Musculoskeletal Research in 2004, which leads me to participate in many top biomedical researches. I would like to thank Dr. William Hamel for his support and provided many resources and help during the research presented in this dissertation. I would like to thank Dr. Aly Fathy in providing his expertise and insights in wireless technologies.

I would also like to thank my fellow researchers at the Center Musculoskeletal Research for their friendship and support, especially Dr. Michael Kuhn, Dr. Brandon Merkl, Dr. Emily Pritchard, Dr. Boyd Evans, Dr. John Mueller, Dr. Filip Leszko, Dr. Joseph Bader, Dr. Shaun Stinton, Dr. Joel Outten, Dr. Adrija Sharma, Joseph Mitchell, Jake Smith, Christopher Carr, Wesley Underwood, Emam Elhak, Admad Algohary, Michael Johnson, Hatem El Dakhakhni, Nicholas Battaglia, Sumesh Zingde, Jon Huber, Katie Anderson, and Mathew Anderle, whom I have worked directly with in many projects. It was a privilege to work and wrestle ideas among all these brilliant individuals. I would also like to give special thanks to Rebecca Robertson, Lyndsay Bowers and Ada Pfohl for their friendship and support. Many researches cannot be done without their hard work. I would like to thank the members from the online community

at edaboard, TI and sparkfun forums, who have given many useful advices in practical hardware implementation.

I would also like to thank my friends, Ernie Fulton, Tabetha Napier, Dr. Wenchao Qu, and Dr. Nicky Lam, and anyone else not mentioned, who have provided me with a lot of personal and emotional support over some of the tougher years.

Abstract

This dissertation examined the inertial tracking technology for robotics and human tracking applications. This is a multi-discipline research that builds on the embedded system engineering, Bayesian estimation theory, software engineering, directional statistics, and biomedical engineering.

A discussion of the orientation tracking representations and fundamentals of attitude estimation are presented briefly to outline the some of the issues in each approach. In addition, a discussion regarding to inertial tracking sensors gives an insight to the basic science and limitations in each of the sensing components.

An initial experiment was conducted with existing inertial tracker to study the feasibility of using this technology in human motion tracking. Several areas of improvement were made based on the results and analyses from the experiment. As the performance of the system relies on multiple factors from different disciplines, the only viable solution is to optimize the performance in each area. Hence, a top-down approach was used in developing this system.

The implementations of the new generation of hardware system design and firmware structure are presented in this dissertation. The calibration of the system, which is one of the most important factors to minimize the estimation error to the system, is also discussed in details. A practical approach using sequential Monte Carlo method with hyper-dimensional statistical geometry is taken to develop the algorithm for recursive estimation with quaternions.

An analysis conducted from a simulation study provides insights to the capability of the new algorithms. An extensive testing and experiments was conducted

with robotic manipulator and free hand human motion to demonstrate the improvements with the new generation of inertial tracker and the accuracy and stability of the algorithm. In addition, the tracking unit is used to demonstrate the potential in multiple biomedical applications including kinematics tracking and diagnosis instrumentation.

The inertial tracking technologies presented in this dissertation is aimed to use specifically for human motion tracking. The goal is to integrate this technology into the next generation of medical diagnostic system.

Contents

1. Introduction 1
1.1 Motivation 1
1.2 Background in Motion Tracking Technology 2
1.2.1 Optical Tracking Technology 2
1.2.2 Electromagnetic (EM) motion tracking technology 4
1.2.3 Global and Radio frequency (RF) Local Positioning System 5
1.2.4 Inertial motion tracking technology 6
1.2.5 In-vivo tracking technologies 8
1.2.6 Human motion tracking Using Inertial Tracking Technologies 10
1.3 Contributions 13
1.3.1 Quaternion estimation with sequential Monte Carlo method (Particle Filter) and directional statistics for Non Gaussian, Non linear system 13
1.3.2 High tracking accuracy for biomedical applications 14
1.3.3 Modular system design 15
1.4 Organization 16
2. Attitude Representations and Estimation Algorithms 18
2.1 Overview of Orientation Representations 18
2.1.1 Rotation Matrix 18
2.1.2 Euler Angle Representation 20
2.1.3 Axis-Angle Representations 21
2.1.4 Euler-Rodrigues symmetric parameters 22

2.1.5 Quaternion 23
2.1.6 Cayley-Klein Parameters 24
2.1.7 Rodrigues Parameters 25
2.1.8 Modified Rodrigues Parameters 25
2.2 Review of Attitude Estimation Algorithm 26
2.2.1 Recursive Bayesian Estimation 27
2.2.2 Principle of Kalman Filter 27
2.2.3 Principle of Extended Kalman Filter 31
2.2.4 Kalman and Extended Kalman filter with QUEST 33
2.2.5 Principle of Unscented Kalman Filter 35
2.2.6 Sequential Monte Carlo Methods 38
2.2.7 Principles of Complementary filter 43
3. Operating Principle of Inertial Sensors 45
3.1 Inertial Measurement Unit Reference Axes and Frames 45
3.2 Measuring Principle of Accelerometer 46
3.3 Measuring Principle of Gyroscopes 50
3.4 Measuring Principle of Magnetometers 52
3.5 Types of Inertial Measurement Units 54
3.5.1 Gimbaled Inertial Platform 54
3.5.2 Strapdown Inertial Measurement Unit System 55
4. Pilot experiment 57
4.1 Off the Shelf IMU specification for pilot experiment 57

4.2 Accelerometer calibration 58
4.3 Gyroscopes Calibration 60
4.4 Magnetometer Calibration 62
4.5 Attitude Estimation Experiment 63
4.6 Human body motion tracking 65
5. Hardware implementation 73
5.1 Hardware specification 73
5.2 Power Supply Design 75
5.3 Signal conditioning circuit Design 75
5.4 Electronics Design 78
5.4.1 Firmware Design 78
5.5 Modular IMU design 83
5.5.1 Circuit layout 84
6. Software Implementation 86
6.1 Kalman Filter Implementation 86
6.2 Extended Kalman Filter Implementation 89
6.3 Unscented Kalman Filter 92
6.4 Complementary filter implementation 93
6.5 Sequential Monte Carlo Methods Implementation 94
6.5.1 Directional Statistic 95
6.5.2 Stiefel Manifold 95
6.5.3 von Mises- Fisher Distribution 96

6.5.4 Bingham Distribution 100
6.5.5 Sequential Monte Carlo Methods with von Mises-Fisher Density ... 104
6.5.6 Sequential Monte Carlo Methods with non-Uniform Density 111
6.5.7 Sequential Monte Carlo Methods with Bias compensation 112
7. Experimental Results 114
7.1 Synthetic Data Simulation 114
7.2 Robotic application 120
7.3 Human motion tracking 129
7.4 System analysis 141
8. Biomedical Applications 145
8.1 Knee joint activities 145
8.2 Lumbar spine activities 152
9. Conclusion 163
10. Future work 165
10.1 Hardware 165
10.2 Calibration 166
10.3 Algorithm Design 166
10.4 Potential Applications 167
A. Appendix A 178
B. Appendix B 186
C. Appendix C 191
D. Appendix D 193
Vita 196

List of Tables

Table 1-1 - Comparison between different motion tracking systems 12

Table 2-1 - Rotation sets for Euler Angle 20

Table 4-1 - Specification of IMU used in pilot study 57

Table 4-2 – RMSE on each axis during static and dynamic testing 68

Table 4-3 - Resolving power of the lease significant bit on each sensor 71

Table 5-1 - Inertial sensors used in prototype 74

Table 5-2 - Routing table from Microcontroller 79

Table 6-1 - Pseudo code for quaternionic Kalman Filter algorithm 88

Table 6-2 - Pseudo code for quaternionic Extended Kalman Filter algorithm 91

Table 6-3 - Pseudo code for von Mises distribution simulation 98

Table 6-4 - Pseudo code for Non-Uniform distribution simulation 102

Table 6-5 - Pseudo code for expectation calculation for quaternion 108

Table 7-1 – Testing Algorithms for the experiment 114

Table 7-2 - RMSE of the testing algorithms for synthetic signal 117

Table 7-3 - RMSE of the testing algorithms for static testing (Robotic) 124

Table 7-4 - RMSE of the testing algorithms for dynamic testing (Robotic) 128

Table 7-5 - RMSE of the testing algorithms for static testing (Free hand) 137

Table 7-6 - RMSE of the testing algorithms for dynamic testing (Free hand) 140

Table 7-7 – Comparison orientation accuracy between OTS and Modular IMU 142

Table 8-1 - RMSE of the testing algorithms for dynamic testing (thigh) 150

Table 8-2 - RMSE of the testing algorithms for dynamic testing (shank)..... 151

Table 10-1 – Summary on future work 168

Table D-1- Pseudo code of Deterministic Resampling Technique [99] 193

Table D-2- Pseudo code of Residual Resampling Technique [66]..... 194

Table D-3- Pseudo code of Auxiliary Resampling Technique [100] 195

List of Figures

Figure 1-1: Polaris Spectra optical tracking system from Northern Digital Inc. (a) optical receiver, (b) passive optical probe with four reflective markers [5]. 3

Figure 1-2 - Passive marker optical tracking system from Motion Analysis Corporation showing the capability of motion capture during ambient sunlight [6] 4

Figure 1-3: NDI Polaris Spectra view volume with 2.050 m in the z-direction and an in-plane area of 1.856 x 1.470 m2 at a z distance of 3.000 m [5] 4

Figure 1-4 - Magnetic tracking system from Acension Technology [12] 5

Figure 1-5 - IMU by Xsens (Left) [21] and MicroStrain (Right) [22] 7

Figure 1-6 - Patient performing a deep knee bend activity under fluoroscopic surveillance 8

Figure 1-7 - 2D image to 3D model registration 8

Figure 1-8 - RSA tracking of a hip implant [30] 10

Figure 2-1 - Rotation of Object A around $\boldsymbol{az1}$ 18

Figure 2-2 - Principle of Complementary Filter ($\boldsymbol{FLPFs}$ is a low pass filter) 43

Figure 2-3 - Principle of Complementary Filter for IMU system 44

Figure 3-1 - Reference frames and axes of the IMU system 45

Figure 3-2 - Conceptual illustration of accelerometers 47

Figure 3-3 - Illustration of the micro-cantilever to the effect of gravity. Gravity has full effect on the microcantilever in a, and no effect in c. 49

Figure 3-4 - Principle of gyroscope 51

Figure 3-5 - Operating principle of vibratory gyroscope 51

Figure 3-6 - Principle of magnetometer. The dash arrows represent the direction of the magnetic field H, and angle $\boldsymbol{\theta}$ is the deviation angle between the coil and the magnetic field 53

Figure 3-7 - Gimballed inertial platform by Marconi Electronics (now BAE systems) shows the instrumentation are surrounded by 3 gimbal rings 55

Figure 3-8 - GG1320 ring laser gyros for military aircraft inertial system [71] 56

Figure 3-9 - MEMS based strapdown IMU by Sparkfun Electronic 56

Figure 4-1 - Data transmission protocol 57

Figure 4-2 - Plastic arm for accelerometer calibration 59

Figure 4-3 - Raw data from the accelerometer 59

Figure 4-4 - Readout of the accelerometer tilting around the x axis 59

Figure 4-5 - Calibrated static output 60

Figure 4-6 - Calibrated tilt angle output of Y and Z around the x axis 60

Figure 4-7 - raw data from the gyroscopes 61

Figure 4-8 - Raw data of the gyroscope at different turning rate of the turn table around x axis 61

Figure 4-9 - Deviation angle between the sensors to true north calculated from magnetometer rotating around x-axis 62

Figure 4-10 - Kalman filter models for IMU 63

Figure 4-11 - Output after Kalman filter 64

Figure 4-12 - Output after EKF 64

Figure 4-13 - IMU brace 65

Figure 4-14 - Static testing with IMU brace. (a) Standing still; (b) Sitting still 65

Figure 4-15 - IMU vs optical outputs (Standing, Thigh) Light blue: Optical X, Purple: Optical Y, Yellow: Optical Z; Dark blue: IMU X, Green: IMU Y, Red: IMU Z 66

Figure 4-16 - IMU vs optical outputs (Standing, Shank) Dark blue: Optical X, Green: Optical Y, Red: Optical Z; Purple: IMU X, Light blue: IMU Y, Yellow: IMU Z 66

Figure 4-17 - IMU vs optical outputs (Sitting, Thigh) Light blue: Optical X, Purple: Optical Y, Yellow: Optical Z; Dark blue: IMU X, Green: IMU Y, Red: IMU Z 67

Figure 4-18- IMU vs optical outputs (Sitting, Shank) Dark blue: Optical X, Green: Optical Y, Red: Optical Z; Purple: IMU X, Light blue: IMU Y, Yellow: IMU Z 67

Figure 4-19 - Deep knee bend sequence 68

Figure 4-20 - Chair rise sequence 68

Figure 4-21 - IMU vs optical outputs (Deep Knee Bend, Thigh). Dark blue: Optical X, Green: Optical Y, Red: Optical Z; Light blue: IMU X, Purple: IMU Y, Yellow: IMU Z..... 69

Figure 4-22 - IMU vs optical outputs (Deep Knee Bend, Shank). Dark blue: Optical X, Green: Optical Y, Red: Optical Z; Purple: IMU X, Light blue: IMU Y, Yellow: IMU Z..... 69

Figure 4-23 - IMU vs optical outputs (Chair RiseThigh). Dark blue: Optical X, Green: Optical Y, Red: Optical Z; Light blue: IMU X, Purple: IMU Y, Yellow: IMU Z 70

Figure 4-24 - IMU vs optical outputs (Chair Rise, Shank). Dark blue: Optical X, Green: Optical Y, Red: Optical Z; Purple: IMU X, Light blue: IMU Y, Yellow: IMU Z 70

Figure 4-25 - Box plot of the X-axis of the gyroscopes undergoing rotations at different angular velocities 72

Figure 5-1 Box plot of the X-axis of the gyroscopes (LPR510-AL) undergoing rotations at different angular velocities 73

Figure 5-2 - System Architecture for Prototype IMU 74

Figure 5-4 - Signal conditioning circuit for accelerometers and gyroscopes 76

Figure 5-5 - Circuit simulation for the signal conditioning circuit 76

Figure 5-6 – Signal conditioning circuit for Magnetometers 77

Figure 5-7 - Firmware instruction flow for end devices (A) 80

Figure 5-8 - Firmware instruction flow for end devices (B) 81

Figure 5-9 -Firmware instruction flow for Access Point 82

Figure 5-10 – Wireless transmission data format 83

Figure 5-11 - The evolution of the IMU circuit designs 84

Figure 5-12 – The sensing strips and electronic board of the modular IMU design (A). The assembled IMU system (B) 85

Figure 6-1- Generic 3D distribution 93

Figure 6-2 – Randomly sampled quaternions with von Mises-Fisher distributions with different dispersion factor. The samples are projected to 3-sphere with identity matrix. The red lines indicate the mean direction of the samples. 99

Figure 6-3 – Randomly sampled quaternions with non-uniform distributions with different density proportion in *K*. The samples are projected to 3-sphere with identity matrix. The red lines indicate the mean direction of the samples. 103

Figure 6-4 – Functional block diagram of the PF algorithm with von Mises-Fisher density 105

Figure 6-5 – Estimation function between particles uncertainty and dispersion factor . 106

Figure 6-6 - Receiving time lag between samples 111

Figure 6-7 - Functional block diagram of the PF algorithm with non-uniform density . 113

Figure 7-1 - Synthetic signal simulated raw output of the sensors 115

Figure 7-2 – Error between different algorithms and the intended motion 116

Figure 7-3 – RMSE of the testing algorithms for synthetic signal 116

Figure 7-4 – Average processing time of the testing algorithms 118

Figure 7-5 - Difference in Rejection Rate for PF with (Left) and without (Right) BC. Magenta indicates rejected particles 119

Figure 7-6 – Experimental setup with robotic manipulator 120

Figure 7-7 – Raw sensor output showing the magnetic interference of the robot to the magnetometer 121

Figure 7-8 – RMSE of PF without BC against benchmark algorithms in static experiment 123

Figure 7-9 – RMSE of PF with BC against benchmark algorithms for static experiment 123

Figure 7-10 – Comparison between IMU data reconstructed with PF with the combination of NU density, BC and 256 particles population and the optical data. 125

Figure 7-11 – RMSE of PF without BC against benchmark algorithms in dynamic experiment 126

Figure 7-12 – RMSE of PF with BC against benchmark algorithms in dynamic experiment 127

Figure 7-13 – Plastic container for both the IMU and optical systems for free hand motion 129

Figure 7-14 – Free hand motion testing 130

Figure 7-15 – Signal variance of the IMU tracker during still, static (robot), static (human) 132

Figure 7-16 – One of the activities showing the instability of CF during human motion testing 132

Figure 7-17 – Activity in Figure 7-16 processed with PF (NU density, BC, RES, 256 particles) 133

Figure 7-18 – Activity showing the instability of EKF during human motion testing 134

Figure 7-19 – Activity in Figure 7-18 processed with PF (NU density, BC, RES, 256 particles) 134

Figure 7-20 – RMSE of PF without BC against benchmark algorithms in static experiment 135

Figure 7-21 – RMSE of PF with BC against benchmark algorithms in static experiment 136

Figure 7-22 – RMSE of PF without BC against benchmark algorithms in dynamic experiment 138

Figure 7-23 – RMSE of PF with BC against benchmark algorithms in dynamic experiment 139

Figure 7-24 – Average processing time of the testing algorithms for free hand activities 141

Figure 7-25 – Raw magnetometers output (Blue: Z axis, Red: Y axis, Orange: X axis) ... 143

Figure 7-26 – Corrected magnetometers output (Blue: Z axis, Red: Y axis, Orange: X axis) 143

Figure 8-1 – Knee joint dynamic study set up. The coordinate system between the IMU tracker are shown in the left hand side of the figure. 146

Figure 8-2 –Subject performing deep knee bend activities 147

Figure 8-3 – Subject performaning chair rise activities 147

Figure 8-4 — Comparison between the RMSE of PF and the benchmark algorithms for 148

Figure 8-5 — Comparison between the RMSE of PF and the benchmark algorithms for 149

Figure 8-6 — Average processing time of all testing algorithms for knee joint activities 149

Figure 8-7 – IMUs setup for lumbar spine experiment 153

Figure 8-8 – Subject performing flexion / extension activity 153

Figure 8-9 – Subject performing lateral bending activity 154

Figure 8-10 – Subject performing axial rotation activity 154

Figure 8-11 – Orientations of IMU located at L1 relative to the reference inertial frame for healthy subject 155

Figure 8-12 – Orientations of IMU located at at L5 relative to the reference inertial frame for healthy subject 155

Figure 8-13 – Orientations of IMU located at L1 relative to the reference inertial frame for degenerative subject 156

Figure 8-14 – Orientations of IMU located at L5 relative to the reference inertial frame for degenerative subject 156

Figure 8-15 – Method to calculate the absolute orientation difference between the L1 IMU and L5 IMU axis in the L5 IMU frame 157

Figure 8-16 – Absolute change in orientation of each axis of L1 relative to L5 159

Figure 8-17 – Absolute change in orientation of each axis of L1 relative to L5 159

Figure 8-18 – Absolute change in orientation of each axis of L1 relative to L5 160

Figure 8-19 – Absolute change in orientation of each axis of L1 relative to L5 160

Figure 8-20 – Absolute change in orientation of each axis of L1 relative to L5 161

Figure 8-21 – Absolute change in orientation of each axis of L1 relative to L5 161

Figure 8-22 – Empty space between the IMU and the back of the subject 162

Figure A-1 - Schematic for powering circuit 178

Figure A-2: Schematic for Microcontroller and JTAG setup 179

Figure A-3: Schematic for ADC and clock 180

Figure A-4: Schematic for Accelerometers and its signal conditioning circuit (1.5g setup) 181

Figure A-5: Schematic for Accelerometer and its signal conditioning circuit (6g setup) 182

Figure A-6: Schematic for Gyroscopes and its signal conditioning circuit (+/- 100 dps setup) 183

Figure A-7: Schematic for Magnetometer its signal conditioning circuit and set/reset trigger 184

Figure A-8: Schematic for wireless transmitter interface 185

Figure B-1: Layout for signal processing board (Top Copper) 186

Figure B-2: Layout for magnetometer sensor strip (Top Copper) 186

Figure B-3: Layout for signal processing board (Bottom Copper) 187

Figure B-4: Layout for magnetometer sensor strip (Bottom Copper) 187

Figure B-5: Layout for signal processing board (Inner Layer: Power) 188

Figure B-6: Layout for magnetometer sensor strip ((Inner Layer: Power) 188

Figure B-7: Layout for signal processing board (Inner Layer: Ground) 189

Figure B-8: Layout for magnetometer sensor strip ((Inner Layer: Ground) 189

Figure B-9: Layout for accelerometers and gyroscopes sensor strip (Top Copper) 190

Figure B-10: Layout for accelerometers and gyroscopes sensor strip (Bottom Copper). 190

Figure C-1: Expectations between quaternion and MRP converted quaternion; (Left) no offsets, (Right) offset rotation [45 45 45] 191

Figure C-2: Comparsion between the expectations of the original quaternion sampled from vMF density and the quaternion converted from the MRP expectation; (Left) offset rotation: [0 4 359], (Right) offset rotation: [355 0 5] 192

Figure C-3: Comparsion between the expectations of the original quaternion sampled from NU density and the quaternion converted from the MRP expectation; (Left) offset rotation: [0 4 359], (Right) offset rotation: [355 0 5] 192

Acronyms

2D	Two-dimensional
3D	Three-dimensional
ADC	Analog to Digital Converter
AHRS	Attitude Heading References System
AP	Access Point
AP	Anterior-Posterior
AUX	Auxiliary Resampling
BC	Bias Compensation
BOD	Beginning of Data
CCD	Charge-Coupled Device
CF	Complementary Filter
DET	Deterministic Resampling
DOF	Degrees of Freedom
ED	End Device
EKF	Extended Kalman Filter
EM	Electromagnetic
ENOB	Effective Number of Bit
EOD	End of Data
FPGA	Field Programmable Gate Array
GPS	Global Positioning System
IMU	Inertial Measurement Unit
KF	Kalman Filter
L1	Lumbar spine vertebrae body 1
L5	Lumbar spine vertebrae body 5
LPF	Low Pass Filter
LPS	Local Positioning System
LSB	Least Squared Bit
MCU	Microcontroller
MEMS	Micro-electro-mechanical systems
ML	Medial-Lateral
MRP	Modified Rodrigues Parameter
NU	Non-Uniform
OTS	Off the shelf
pdf	Probability Density Function
PF	Particle Filter
QUEST	QUaternion ESTimator
REQUEST	Recursive QUaternion ESTimator
RES	Residual Resampling
RF	Radio Frequency
RLG	Ring Laser Gyroscope
RMSE	Root mean squared error
RSA	Roentgen stereophotogrammetry analysis
RX	Receiver
SI	Superior-Inferior
SIS	Sequential Importance Sampling
SLERP	Spherical Linear Interpolation
SMC	Sequential Monte Carlo
SO(3)	Special Orthogonal (3D rotation) group
TDMA	Time Division Multi Access
TX	Transmitter
UART	Universal Asynchronous Receiver/Transmitter
UKF	Unscented Kalman Filter
UWB	Ultra Wide Band
vMF	von Mises-Fisher

1. Introduction

1.1 Motivation

Human body motion tracking is an important area of research where many medical applications can be derived based on this technology. Medical device researchers and engineers use motion analysis to study the kinetics and kinematics of the human joint, and to design better implants and prostheses. Motion analysis is also used as diagnostic tools to acquire valuable information for physicians. Furthermore, many surgeries have begun to incorporate robots as part of the operating procedures to improve the overall surgical outcome. The performance of these systems relies heavily on the positioning accuracy of the tracking system of these surgical robots. In addition, motion tracking system is also used in exercise science and sport medicine to improve the performance of the athletes.

The principle of motion tracking is to resolve the locations and orientations of an object in three-dimensional (3D) space during motion. There is a vast variety of tracking systems that utilize different localization and tracking technologies. Tracking technologies can be divided into two main categories, which either observes a certain number amount of fixed points on an object externally or estimating poses between sequential frames temporarily. Optical, electro-magnetic, global positioning system, radio frequency (RF) positioning systems are examples of exterior pose monitoring system. In addition to these technologies, a sub-class of external tracking using radioactive based imaging techniques is often used to track in-vivo motions. Fluoroscopy is one of the frequently used in-vivo tracking systems.

Inertial tracking system, on the other hand, is an example of relative pose estimating system. In recent years, with the advance of semiconductor based inertial

sensing technologies and sophisticated designs in attitude estimation algorithm, inertial tracking system has been used extensively in many different tracking applications. Previous studies discuss in the later section of this dissertation demonstrates the potential of inertial tracking for human body motion. However, the dynamic range of the human motion varies with activities and the motion of interest. The options for off the shelf inertial tracker are limited and significantly hinder the optimal tracking capability. In addition, current algorithm design does not model the change of the signal characteristics and artifacts after the system is attached to the human body. Algorithm stability becomes an issue as the some of the algorithms' assumptions no longer apply.

The motivation of this dissertation is to develop an inertial tracking system for human body motion tracking by optimizing the tracking problem from multiple disciplines.

In order to contextualize the challenges and necessities of human body motion tracking technologies, the section below provides a comprehensive background review on the current tracking technologies and discussion concerning the advantages and disadvantages of each system.

1.2 Background in Motion Tracking Technologies

1.2.1 Optical Tracking Technology

Optical tracking is generally considered to be the standard for body motion tracking. It is typically done with infrared cameras and a set of either active or passive markers. Active markers emit infrared ray signals to the camera, whereas passive markers have reflective coating to reflect the infrared flash emitted adjacent to the camera [1]. A passive marker optical tracking system is shown in Figure 1-1. Infrared camera is the choice for optical tracking as it is outside the visible colour spectrum,

hence reducing the ambiguity between the markers and the background (Figure 1-2). A minimum of two cameras are needed for triangulating and estimating the position of the markers in 3D. The system can only resolve the translation motion on each of the marker. The rotation of the object, however, requires attaching multiple markers to determine the orientation via rigid body dynamics.

The camera in the optical system has a viewing volume that guarantees the object is within the acceptable accuracy as seen in Figure 1-3. Hence, for activities such as gait analysis, multiple cameras are necessary to ensure the markers on the subject to stay within the viewing volume [2]. Optical system also suffers from the line of sight issue. Any marker that is obstructed from the cameras will adversely affect the tracking performance. Multiple camera setup can eliminate the uncertainty of the occluded markers; improving the accuracy of the system. However, such measure significantly increases the cost of the system. In addition, the mobility of the equipment becomes very limited [3]. Optical tracking is known to have the highest accuracy among all tracking device system. The rated 3-D accuracy of the Polaris Spectra system is 0.25 mm root-mean-squared error (RMSE) for a single marker, 0.233 mm RMSE for 3-D position of a rigid body with active markers, and 0.231 mm RMSE for 3-D position of a rigid body with passive markers [4].

Figure 1-1: Polaris Spectra optical tracking system from Northern Digital Inc. (a) optical receiver, (b) passive optical probe with four reflective markers [5].

Figure 1-2 - Passive marker optical tracking system from Motion Analysis Corporation showing the capability of motion capture during ambient sunlight [6]

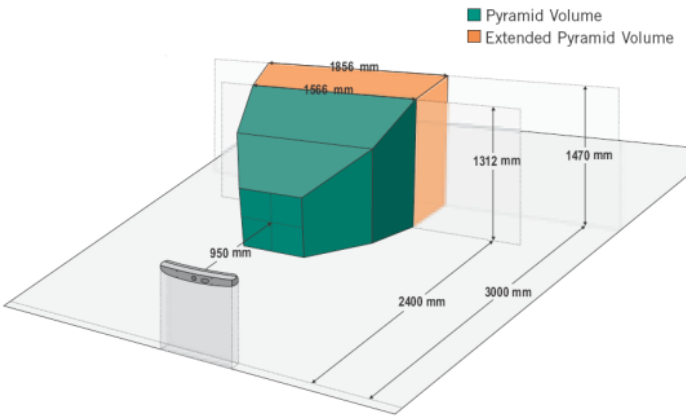

Figure 1-3: NDI Polaris Spectra view volume with 2.050 m in the z-direction and an in-plane area of 1.856 x 1.470 m2 at a z distance of 3.000 m [5]

1.2.2 Electromagnetic (EM) motion tracking technology

Electromagnetic (EM) tracking is done with measuring the magnetic field created from three sets of sensor coils that are placed orthogonal to each other [7], [8]. These coils act as transmitter, and are excited in sequence by a driving circuit to generate EM field. Another set of fixed coils that act as the receiver, monitor the changes of the pulses of the EM field to determine the six degrees of freedom of the transmitter.

A magnetic tracking by Acension Technology is shown in Figure 1-4. Due to the nature of this type of sensor, ferromagnetic materials can disrupt the magnetic field generated by the coils; thus decreasing the accuracy of the system [4,9,10]. Hence, the primary focus in EM tracking research has been focused on calibrating the device, i.e. either eliminate or account for such disruption [11].

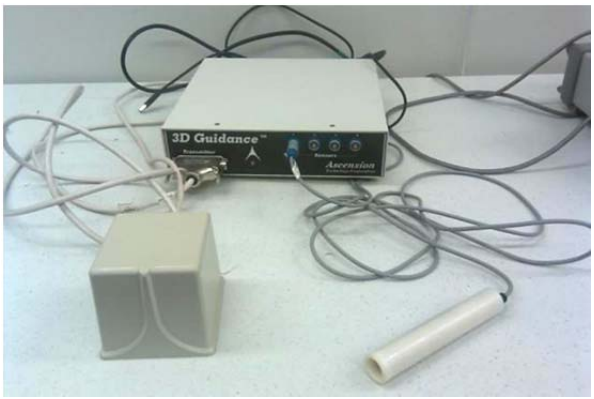

Figure 1-4 - Magnetic tracking system from Acension Technology [12]

The major drawback of EM tracking, aside from the ferromagnetic interference, is the need of to use wires to connect the tracking unit. The 3-D static position accuracy of the system was reported to be 1.8 mm RMSE using the mid-range transmitter over an operating range of 0.203 to 0.762 m from the transmitting coil to the sensor [12]. Dynamic accuracy of the system is not available due to non-reproducible magnetic disturbance during testing. The accuracy of the system also diminishes significantly as the transmitter moved further away from the receiver. The maximum operating range of the system with 1.8 mm RMSE is 0.75 m from transmitting coil to magnetic sensor.

1.2.3 Global and Radio frequency (RF) Local Positioning System

Tracking technique such as global positioning system (GPS) uses multiple satellites for positioning [13]. The sensor transmits a signal to multiple satellites in orbit. The difference of the signal arrival time can be used to triangulate the position of the object. There are multiple error sources in GPS, including Ionospheric effect [14], and Tropospheric effect [15], which are uncommon in other motion tracking system. The accuracy of the GPS is approximately 5 to 30m. It is reliable for geo-localization. However, it's not accurate enough for human motion analysis. GPS is also limited to outdoor only where you can maintain line of sight communications to satellites.

Local positioning system (LPS) uses the same concept as GPS; however, the satellites are set up as base stations and fixed indoor. Multiple LPS technologies have been developed with RF technologies. Recently, ultra-wide-band (UWB) systems using 3.1 to 10.6 GHz channels bandwidth have shown promising results as a local positioning system [16,17]. The current leading edge research in UWB systems have reached sub millimeter static accuracy and less than 5mm in dynamic motion in indoor environment [18]. The mobility of the system is limited as the LPS satellites or base stations are fixed and mounted indoor. Additionally, multi-tag and targets interferences adversely affect the accuracy of the system, and the solution is still being actively researched.

1.2.4 Inertial motion tracking technology

With the limitation of the exterior monitoring tracking systems mentioned above, a different tracking strategy was introduced. Inertial Measurement Unit (IMU) measures the inertial properties of an object in motion; and uses the laws of physics to calculate the relative motion of the object to its initial position. IMU is usually consisted of multiple inertial measuring sensors, e.g. accelerometers and gyroscopes. It is capable of tracking without using any external observer. Using the information from the sensors such as accelerations and angular velocities, the positions and orientations of the object can be estimated relative to its previously known location. This method is known as dead reckoning, which will be discussed in detail in the later chapter. Additional sensors may be added on it to increase the capability and accuracy of the system.

IMU was initially designed by Robert Goddard for rocket stabilization, and later developed into missiles guidance system in World War II [19]. The early development of inertial sensors shown the system was subjected to accumulative drifting error, which severely reduces the accuracy of the system over extended period of time. During that

time, the primary focus of research was to develop more sensitive and accurate sensors to minimize the potential noise in calculation. The performance of the inertial tracking system improved drastically with the advance of digital computing, which replaces the original analog computer. Digital computer also enables the implementation of more complex estimation algorithms. In the early 60s during the Apollo project, the IMU was used as a guidance and navigation for human flight for the first time in history [20]. The use of this technology is now wide spread into other navigation systems for aero-planes, ships, and submarines. In recent years, semiconductor based IMU has been integrated with GPS system to provide navigation for automobiles and unmanned vehicles. GPS provides an accurate checkpoint for correcting the drift in IMU. At the same time, the IMU provides real time positions and orientations in between each GPS updates, as well as during GPS blackout.

There are multiple types of IMU system depending on the accuracy required and the available space. The micro-machined micro-electro-mechanical system (MEMS) based sensors coupled with integrated circuit (IC) allow miniaturization of the IMU to portable system as shown in Figure 1-5. The static accuracy of this system is 0.4 degrees in orientation; whereas the dynamic accuracy under cyclic motion is 4 degrees [21,22].

Figure 1-5 - IMU by Xsens (Left) [21] and MicroStrain (Right) [22]

1.2.5 In-vivo tracking technologies

The optical tracking device are usually used to perform motion tracking outside of the human body, although there are several attempts to use them in-vivo during certain surgical procedures [23,24]. EM tracking device has been used for tracking the needle during injection [25].

1.2.5.1 Fluoroscopy

In-vivo bone tracking is a primary interest to the orthopedic industry. Understanding the biomechanics of the human joint plays an important role in the implant design process. Fluoroscopic is an example of a bio-imaging unit that is capable of performing in-vivo motion analysis. Fluoroscopy unit consists of an X-ray source, an image intensifier or a flat panel detector couple with a charge-coupled device (CCD) camera. The x ray is attenuated as it interacts with different soft tissues of the body. The CCD camera records the image and displays them as video. Figure 1-6 shows a subject performing deep knee bend motion under fluoroscopy. An image processing technique was developed to register the two dimensional (2D) shadow casted from the area of interest to a 3D polygon models on the computer as shown in Figure 1-7.

Figure 1-6 - Patient performing a deep knee bend activity under fluoroscopic surveillance

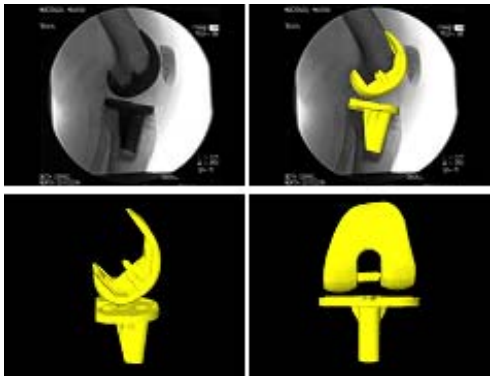

Figure 1-7 - 2D image to 3D model registration

The dynamics of the motion captured by the fluoroscopy unit can be evaluated after registering the models on multiple frames [25,26,27]. Fluoroscopy suffers the same problem as optical device; i.e. the viewing volume is limited. Motion such as walking, jogging can easily go out of range of the viewing volume of the fluoroscopy unit.

1.2.5.2 Roentgen Stereophotogrammetry Analysis

Another approach for in-vivo motion tracking is Roentgen stereophotogrammetry analysis (RSA), which was pioneered by Davidson in the early 19th century [28]. The localization concept is very similar to optical motion tracking using passive markers. Tantalum markers are implanted either onto the bone of the patient or within the prosthesis at known locations. Multiple x-ray images were taken while the patient is performing activities. The measurements of these markers are taken and the poses of the bones or prostheses on each radiograph are calculated as shown in Figure 1-8 [29]. This method is considered quasi static kinematics analysis since the patients are required to hold still when the radiograph is being taken. It has been proven by Fuller that the relative motion between the skin and the bone can introduce a significant amount of error while performing kinematic analysis [30]. While the IMU itself cannot track the motion of the in-vivo joint, it is possible to couple with other non-radioactive imaging techniques such as ultrasound to achieve in vivo tracking. IMU provides the kinematic data between the global and skin frames, while the ultrasound transducers provide the skin to bone transformation. This promotes the concept of a low cost, non-radioactive, high mobility in-vivo motion tracking device that can be used in many areas such as sport science and orthopedics research. In this dissertation, the primary focus is to develop the strategy in maximizing the accuracy of the IMU for human body motion tracking with both hardware and software development.

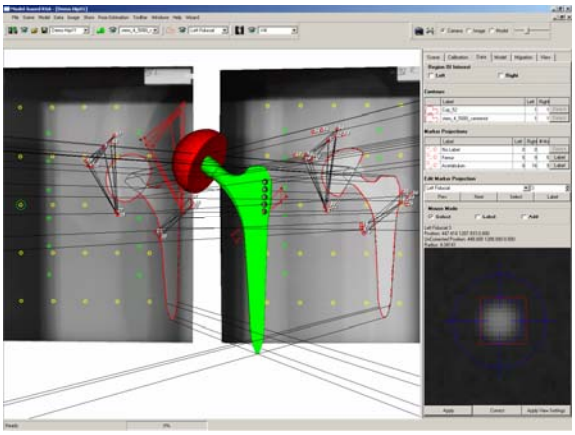

Figure 1-8 - RSA tracking of a hip implant [31]

1.2.6 Human motion tracking Using Inertial Tracking Technologies

The initial effort to use IMU to track human body motion began towards the end of 90s; where the technology to design and fabricate Micro-Electro-Mechanical System (MEMS) based sensors began to mature. In 1996, Veltink published the initial concept of using accelerometer sensors to motion analysis using uniaxial accelerometer [32]. Bouten used triaxial accelerometer to analysis daily activities such as standing, walking, and performing desk work [33]. Multiple studies had been performed for posture estimations and daily activities monitoring using tri-axial accelerometers [34,35,36].

In 1999, Miyazaki measured the stride length and instantaneous velocity of the patients during gait by attaching gyroscopes to their thighs [37], and Tong used gyroscopes for a full gait analysis [38]. In 2002, Mayagoita introduced the initial framework of tracking lower extremities with a fusion of accelerometers and gyroscopes in 2D sagittal plane [39]. Various clinical applications have been proposed using inertial motion tracking for gait analysis [40,41,42], as well as other joint studies [43,44,45]. Roetenberg used a set of tri-axial accelerometers, tri-axial gyroscopes and a magnetometer to monitor human motion in his dissertation in 2006 [46,47]. Many

researchers built on and improved this technology for human motion orientation tracking with IMU [48,49]. There are also multiple researches for incorporating IMU motion tracking into virtual and augmented reality environment [41,50,51].

In recent years, semiconductor based sensor have successfully introduced into the consumer electronics industry. Wii-Remote is one of the most successful commercial systems that incorporate accelerometers, gyroscopes and infrared range sensors for motion tracking as an interface to an entertainment system [52]. The performance of the system mentioned above provides excellent static result, yet the accuracy of dynamic tracking is lacking due to various limitations in the design. This research is aimed to minimize the error from both hardware and software perspectives, and to achieve mili-degree accuracy for human body dynamic motion. Table 1-1 compares all the existing motion tracking technologies and the proposing system in this research.

Table 1-1 - Comparison between different motion tracking systems

	3D Static Translation Accuracy	3D Static Orientation Accuracy	Dynamic Translation Accuracy	Dynamic Orientation Accuracy	Cost	Flexibility (mobility/setup)	Multi-track	In-vivo
Optical	0.25mm (RMSE)	-	0.25mm	-	High	N	Y	N
EM	1.4mm (RMSE)	0.5^{o} (RMSE)	N/A	N/A	Moderate	N	Y	N
GPS	5m	-	5m	-	Low	Y	Y	N
LPS	0.2mm (RMSE)	-	3mm (RMS)	-	Moderate	N	R&D	N
Fluoroscopy	-	-	0.3mm (RMS)	0.3^{o} (RMSE)	High	N	Y	Y
RSA	10–250 μm [53,54]	0.03°–0.06° [53,54]	-	-	Moderate	N	Y	Y
IMU	N/A	$0.4\text{-}1^{o}$ (RMSE)	N/A	4^{o} (cyclic, RMSE)	Low	Y	Y	N
Proposing system	N/A	$<0.5^{o}$ (RMSE)	N/A	$<1^{o}$ (RMSE)	Moderate	Y	Y	N

1.3 Contributions

There are extensive researches in the field of inertial tracking technologies and estimation algorithms. The fundamental contribution presented in this dissertation is realizing a highly accurate inertial tracking system to monitor human body motion with engineering designs based on scientific analysis in multiple disciplines. The following section outlines the contributions presented in this dissertation.

1.3.1 Quaternion estimation with sequential Monte Carlo method (Particle Filter) and directional statistics for Non Gaussian, Non linear system

The analysis of current attitude estimation algorithms leads to the development of the quaternionic attitude estimation algorithm using sequential Monte Carlo method due to the instability issue observed during human motion tracking experiment. Sequential Monte Carlo method (or particle filter) is a sequential approach to a Bayesian estimation problem. It does not assume Gaussian characteristic of the system and it does not require direct determination of the system's states error covariance. This approach allows considerable amount of flexibility in a wide variety of problems. Moreover, it is not necessary to directly determine the true density of the system as the particle filter approximates the statistical properties of the system through sequential importance sampling. The primary challenge is that quaternion is a hyper-complex vector within the special orthogonal group (SO(3)) in the fourth dimension. Statistical inference with quaternion is difficult using conventional statistical tool. Hyper-dimensional directional statistic is used in this dissertation to solve this problem.

In this dissertation, the quaternion particle filter is implemented with two hyper-dimensional statistical geometries, von Mises-Fisher density, and Non-Uniform density. Non-uniform density is based on the principle of Bingham density, and its sampling

technique is presented. The implementation of the particle filter is different between the two densities because of the sampling technique. The merit in this implementation is to realize the likelihood functions in particle filtering for both of the densities. For von Mises-Fisher density, the likelihood is determined through computation of the particle spread to the dispersion factor whereas it is determined by the density bounds evaluation for non-uniform density.

In addition, this dissertation presented a computation method to determine the quaternion expectation via weighted quaternion interpolation using a double recursion with spherical linear interpolation.

1.3.2 High tracking accuracy for biomedical applications

This dissertation presented multiple validation studies to verify the capabilities and potentials of the inertial tracking system and the particle filtering algorithms. A top level simulation model was designed to validate the design of the algorithm. Current simulation model for particle filter design does not capture the signal characteristic of an inertial tracking device. The presented model simulated the outputs from various sensors. The simulation demonstrates the capability of the particle filters, whereas the particles population increases, its accuracy improves. With enough particles, the particle filtering estimation algorithm can outperform the benchmarking algorithms such as the extended Kalman filter and the complementary filter.

The performance of the inertial tracking system and the algorithm were examined with robotic and human free hand motion applications. The performance of the system is comparable with optical tracking system. The experimental results and analyses also indicate that several configurations of the particle filters perform better than the benchmarking algorithms. Human free hand motion activities have also increased the root mean squared error of the estimation significantly compare to the

robotic application. In addition, the particle filter algorithm shows remarkable stability comparing against the benchmarking algorithms during free hand motion activities.

The developed tracking system were used in two biomedical applications to demonstrates its potential. The first application shows the basic multi-tracking ability of the system from multiple knee joint activities, such as deep knee bend and chair rise. The performance of the system is compared with the optical tracking system, and shows great performance with approximately 0.5 degrees root mean squared error on all axes relative to the optical system. The second application shows the capability of the inertial tracking system as a clinical diagnostic device. The tracking system was used on a normal subject and a subject with degenerative lumbar spine. The tracking system monitors the subjects performing several activities including flexing and extending their back and twisting their torso. During these activities, the subject with degenerative spine has more out of plane motions compare to the normal subject. The result from the inertial tracker also correlates with the results found in previous fluoroscopic study.

1.3.3 Modular system design

This dissertation presented the design philosophy and implementation of a modular tracking system. Off the shelf inertial tracking systems utilize sensing elements with a specific dynamic range. However, human body motion goes through phases of activities with different ranges of speed. In order to optimize the performance of the tracking system, the hardware must be able to provide sensing elements in multiple dynamic ranges. A modular design of the inertial tracker was implemented to provide multi-resolution sensing based on different activities. The sensing elements of different properties are organized into different sensing strips that can be plugged in or remove from the base system easily. Secondly, the inertial tracking system presented in this dissertation drastically increases the resolution available for the inertial sensors. The

current high-end inertial tracking system uses 16-bit analog to digital converter. However, the specification generally does not taken account into the performance degradation from conversion speed and signal dynamic range matching. The current design for the modular system provides 18.1 effective noiseless bits matched to the sensors dynamic output range at top conversion speed.

1.4 Organization

There are many orientation representations methods, as well as attitude estimation algorithms. Chapter 2 provides the background information and reviews various types of attitude representations, including Euler angles, axis-angle, Euler Rodrigues symmetric parameters, quaternions, Cayley-Klein parameters, Rodrigues parameters, and modified Rodrigues parameters. The principles of various attitude estimations and tracking theories based on recursive Bayesian estimation, including the Kalman class estimators and its variations and sequential Monte Carlo method, were discussed. Chapter 3 reviews the background and fundamentals of inertial sensing with accelerometers, gyroscopes and magnetometers. The configurations and designs of several inertial tracking devices were also examined. The pilot experiment with an off the shelf IMU is discussed in Chapter 4. The calibration techniques of each of the sensing elements were discussed. The tracking experiment used quaternionic Kalman and Extended Kalman filters to perform attitude estimation. The experimental analysis provides insights to the designs and improvements necessary to achieve high accuracy tracking for human body motion.

The hardware implementation of the new generation of IMU is discussed in Chapter 5. The specification of the system is also discussed. The modular design of the IMU is presented as well as the counter measure to hardware performance degradation

from current off the shelf systems. The signal conditioning circuits for each of the sensor type are shown, and the firmware of the IMU system is illustrated.

In chapter 6, the implementation of various attitude estimation algorithms were presented. The Complementary and Kalman class estimators for quaternion are examined in details. The sequential Monte Carlo methods presented here in this dissertation use hyper-dimensional directional statistical geometries such as von Mises-Fisher and non-uniform densities. The implementations of the sampling methods are shown. In addition, the likelihood functions, the importance sampling and resampling methods for each of the densities are also demonstrated and discussed in details.

In chapter 7, the results from multiple validation experiments are discussed. The validation study for the algorithm designs were conducted through simulation studies. A top-level simulation model for inertial tracking system was presented. The simulation results are compared to the noise free model. A robotic testing was conducted to access the dynamic capabilities of the IMU system and the results is compared to other benchmarking algorithms such as extended Kalman filter and complementary filter relative to the optical tracking system. Free hand motion experiments are also presented to demonstrate the robustness of the particle filter algorithm design. In chapter 8, the inertial tracking systems were tested on the knee joint and lumbar spine to demonstrate the potential as a diagnostic device for biomedical applications. The knee joint activities demonstrate the feasibility of using multiple IMUs for kinematic studies. The performance of the IMU is compared to optical tracking system. The second application demonstrates the feasibility to identify subject with either normal or degenerative lumbar spine by analyzing the orientation result from the IMU during activities. The conclusion of this dissertation is presented in chapter 9, and the discussion related to the future work is discussed in chapter 10.

2. Attitude Representations and Estimation Algorithms

2.1 Overview of Orientation Representations

A six degree-of-freedom (DOF) motion tracker constitutes two sets of data regarding to the position (translations) and orientation (rotations) of an object. The position of an object is commonly represented by a vector in Cartesian, cylindrical, or spherical co-ordinate system. Orientation, on the other hand, has many different kinds of representations, which can be parameterized to sets of vectors or matrixes. The following sections discuss the pros and cons of different parameterization methods for orientation.

2.1.1 Rotation Matrix

Suppose the object A has an orthogonal base axis of $\{\overline{a_x^1}, \overline{a_y^1}, \overline{a_z^1}\}$ and it is rotated by an angle of θ (rad) around $\overline{a_z^1}$ as shown in Figure 2-1 . The rotational transformations of A to its new orientation $\{\overline{a_x^2}, \overline{a_y^2}, \overline{a_z^2}\}$ are given by equations (1-3).

$$\overline{a_x^2} = \overline{a_x^1}cos\theta + \overline{a_y^1}sin\theta \quad (1)$$

$$\overline{a_y^2} = -\overline{a_x^1}sin\theta + \overline{a_y^1}cos\theta \quad (2)$$

$$\overline{a_z^2} = \overline{a_z^2} \quad (3)$$

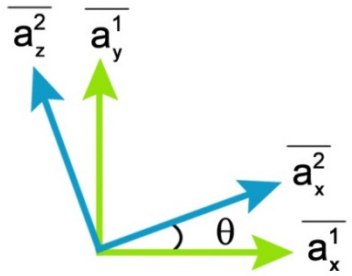

Figure 2-1 - Rotation of Object A around $\overline{a_z^1}$

These are typically expressed in matrix form, which is known as the transformation matrix.

$$\begin{bmatrix} \overline{a_x^2} \\ \overline{a_y^2} \\ \overline{a_z^2} \end{bmatrix} = \begin{bmatrix} cos\theta & sin\theta & 0 \\ -sin\theta & cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \overline{a_x^1} \\ \overline{a_y^1} \\ \overline{a_z^1} \end{bmatrix} \tag{4}$$

Equation (4) demonstrates the transformation of an object rotating around a single axis. To obtain the full three DOF of rotations, the transformations for $\overline{a_x^1}$ in equation (5) and $\overline{a_y^1}$ in equation (6) are required. The orthogonal base axes of the object can then sequentially transformed by these matrixes.

$$\begin{bmatrix} \overline{a_x^3} \\ \overline{a_y^3} \\ \overline{a_z^3} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\rho) & sin(\rho) \\ 0 & -sin(\rho) & cos(\rho) \end{bmatrix} \begin{bmatrix} \overline{a_x^2} \\ \overline{a_y^2} \\ \overline{a_z^2} \end{bmatrix} \tag{5}$$

$$\begin{bmatrix} \overline{a_x^4} \\ \overline{a_y^4} \\ \overline{a_z^4} \end{bmatrix} = \begin{bmatrix} cos(\varphi) & 0 & -sin(\varphi) \\ 0 & 1 & 0 \\ sin(\varphi) & 0 & cos(\varphi) \end{bmatrix} \begin{bmatrix} \overline{a_x^3} \\ \overline{a_y^3} \\ \overline{a_z^3} \end{bmatrix} \tag{6}$$

where ρ is the rotation of object A around $\overline{a_x^2}$, and φ is the rotation around $\overline{a_y^3}$.

The combined rotation matrix (R) is given by,

$$R = \begin{bmatrix} c(\varphi)c(\theta) & -c(\rho)s(\theta) + s(\rho)s(\varphi)c(\theta) & s(\rho)s(\theta) + c(\rho)s(\varphi)c(\theta) \\ c(\varphi)s(\theta) & c(\rho)c(\theta) + s(\rho)s(\varphi)s(\theta) & -s(\rho)c(\theta) + c(\rho)s(\varphi)s(\theta) \\ -s(\varphi) & s(\rho)s(\varphi) & c(\rho)c(\theta) \end{bmatrix} \tag{7}$$

where $c \rightarrow cos, s \rightarrow sin$

Lastly, R has 9 components to represents 3 rotation. Hence, six constraints are required, which can be expressed by

$$R^T R = I, \det(R) = 1 \tag{8}$$

2.1.2 Euler Angle Representation

The angles (ρ, φ, θ) in equation (7) are commonly referred as the Euler angles representation of rotations. There are multiple ways to convert the Euler angles representation into rotation matrix depending on the multiplication sequence of the matrixes. There are a total of six symmetric and six asymmetric sequences as summarized in Due to these limitations, the choice of the sequence of rotations becomes very important. The obvious advantage of using Euler angles is that they are intuitive representation of the rotations. One of the major issues is that the inverse conversion between the rotation matrix and the Euler angles as shown in equations (12-14).

Table 2-1 .

Euler angles are not unique as the mirror angles result as the same orientation as shown in equation (9) for symmetric sets and equation (10) for asymmetric sets

$$R(\rho, \varphi, \theta) = R(\rho + \pi, -\varphi, \theta - \pi) \tag{9}$$

$$R(\rho, \varphi, \theta) = R(\rho + \pi, \pi - \varphi, \theta - \pi) \tag{10}$$

Because of these ambiguities, the following constraints are commonly used to ensure the uniqueness of the Euler angles.

$$0 \leq \rho < 2\pi, \qquad \frac{-\pi}{2} \leq \varphi \leq \frac{\pi}{2}, \qquad 0 \leq \theta < 2\pi \tag{11}$$

Due to these limitations, the choice of the sequence of rotations becomes very important. The obvious advantage of using Euler angles is that they are intuitive representation of the rotations. One of the major issues is that the inverse conversion between the rotation matrix and the Euler angles as shown in equations (12-14).

Table 2-1 - Rotation sets for Euler Angle

Symmetric sets	Asymmetric sets
X-Y-X	X-Y-Z
Y-Z-Y	Y-Z-X
Z-X-Z	Z-X-Y
X-Z-X	X-Z-Y
Y-X-Y	Y-X-Z
Z-Y-Z	Z-Y-X

Singularity occurs at $\varphi = \pm\frac{\pi}{2}$ for equations (123) and ((14) as the denominator is zero. This is known as the mathematical gimbal lock, where two of the rotational axes are parallel to each other.

$$\varphi = \text{asin}(R_{31}) \tag{12}$$

$$\rho = atan\left(\frac{R_{32}}{R_{33}}\right) \tag{13}$$

$$\theta = atan\left(\frac{R_{21}}{R_{11}}\right) \tag{14}$$

2.1.3 Axis-Angle Representations

Consider the rotation of object A is parameterized by a single rotation (α) around a unit length axis ($\hat{e} = [\hat{e}_1 \quad \hat{e}_2 \quad \hat{e}_3]$) in the plane of rotation. There are many different parameterization methods based on this principle. Equation (15) demonstrated the simplest parameterization method with Euler angles (ρ, φ, θ) .

$$\begin{bmatrix} \hat{e}_1 \\ \hat{e}_2 \\ \hat{e}_3 \\ \alpha \end{bmatrix} = \begin{bmatrix} \cos(\rho)\cos(\varphi) \\ \cos(\rho)\sin(\varphi) \\ \sin(\rho) \\ \theta \end{bmatrix} \tag{15}$$

Similar to Euler angles, the parameterization can create different sets of axis-angle representations depending on the choice of the angles used in the above equation. The rotation matrix for axis angle representation is shown in equation (16).

$$R_{\hat{e}} = \begin{bmatrix} \hat{e}_1{}^2(\mathrm{E}) + \cos(\alpha) & \hat{e}_1\hat{e}_2(\mathrm{E}) + \hat{e}_3\sin(\alpha) & \hat{e}_1\hat{e}_3(\mathrm{E}) - \hat{e}_2\sin(\alpha) \\ \hat{e}_2\hat{e}_1(\mathrm{E}) - \hat{e}_3\sin(\alpha) & \hat{e}_2{}^2(\mathrm{E}) + \cos(\alpha) & \hat{e}_2\hat{e}_3(\mathrm{E}) + \hat{e}_1\sin(\alpha) \\ \hat{e}_3\hat{e}_1(\mathrm{E}) + \hat{e}_2\sin(\alpha) & \hat{e}_3\hat{e}_2(\mathrm{E}) - \hat{e}_1\sin(\alpha) & \hat{e}_3{}^2(\mathrm{E}) + \cos(\alpha) \end{bmatrix} \quad (16)$$

where $\mathrm{E} = 1 - \cos(\alpha)$.

2.1.4 Euler-Rodrigues symmetric parameters

The Euler-Rodrigues symmetric parameters ($\eta = [\eta_1\ \eta_2\ \eta_3\ \eta_4]$) are one of the frequently used parameterization methods for axis-angle representation. It is defined as,

$$\begin{bmatrix} \eta_1 \\ \eta_2 \\ \eta_3 \end{bmatrix} = \begin{bmatrix} \hat{\eta}_1\sin(\frac{\alpha}{2}) \\ \hat{\eta}_2\sin(\frac{\alpha}{2}) \\ \hat{\eta}_3\sin(\frac{\alpha}{2}) \end{bmatrix},\ \eta_4 = \cos(\frac{\alpha}{2}) \quad (17)$$

with unity constraint of

$$\eta_1{}^2 + \eta_2{}^2 + \eta_3{}^2 + \eta_4{}^2 = 1 \quad (18)$$

The rotation matrix for Euler-Rodrigues symmetric parameters is shown below.

$$R_\eta = \begin{bmatrix} \eta_1{}^2 - \eta_2{}^2 - \eta_3{}^2 - \eta_4{}^2 & 2(\eta_1\eta_2 + \eta_4\eta_3) & 2(\eta_1\eta_3 - \eta_4\eta_2) \\ 2(\eta_2\eta_1 - \eta_4\eta_3) & -\eta_1{}^2 + \eta_2{}^2 - \eta_3{}^2 + \eta_4{}^2 & 2(\eta_2\eta_3 + \eta_4\eta_1) \\ 2(\eta_3\eta_1 + \eta_4\eta_2) & 2(\eta_3\eta_2 - \eta_4\eta_1) & -\eta_1{}^2 - \eta_2{}^2 + \eta_3{}^2 + \eta_4{}^2 \end{bmatrix} \quad (19)$$

Rotation transformation between two Euler Rodrigues symmetric parameters can be performed in equation (20).

$$R_\eta R_\zeta = \eta \otimes \zeta = \begin{bmatrix} \zeta_1 & -\zeta_2 & -\zeta_3 & -\zeta_4 \\ \zeta_2 & \zeta_1 & \zeta_4 & -\zeta_3 \\ \zeta_3 & -\zeta_4 & \zeta_1 & \zeta_2 \\ \zeta_4 & \zeta_3 & -\zeta_2 & -\zeta_1 \end{bmatrix} \begin{bmatrix} \eta_1 \\ \eta_2 \\ \eta_3 \\ \eta_4 \end{bmatrix} \quad (20)$$

The advantage of Euler Rodrigues symmetric parameters is that it uses considerably less elements compare to the rotation matrix constructed from Euler angles. In addition, it has only one simple constraint as shown in equation (18).

2.1.5 Quaternion

The Euler-Rodrigues symmetric parameters are a special case of a general hyper-complex number group known as quaternion. Quaternion is defined as a vector with one real component q_0 and three imaginary vector component $(q_1 i, q_2 j, q_3 k)$,

$$q = q_0 + q_1 i + q_2 j + q_3 k \quad (21)$$

with the following properties,

$$i^2 = j^2 = k^2 = -1 \quad (22)$$
$$ij = -ji = k \quad (23)$$
$$jk = -kj = i \quad (24)$$
$$ki = -ik = j \quad (25)$$

The conjugate of a quaternion q^* gives the inverse rotation of the Euler-Rodrigues symmetric parameters.

$$q^* = q_0 - q_1 i - q_2 j - q_3 k \quad (26)$$

Quaternions are non-commutative and follow the multiplication rule in equation (20). The inverse of a quaternion is given by the conjugate divided by the norm.

$$q^{-1} = \frac{q^*}{{q_0}^2 + {q_1}^2 + {q_2}^2 + {q_3}^2} \quad (27)$$

Quaternion with norm root equals to one is known as unit quaternion. Unit quaternion uses the same Lie algebra as the special orthogonal rotation group (SO(3)). Each elements in the SO(3) is composed of two unit quaternions $(q, -q)$. Because of this

unique property, rotation representations parameterized as quaternion do not suffer from the mathematical gimbal locking issues in Euler angle representation. Quaternion can be calculated from Euler angles via equations (28-31). Since there are 12 different rotational sets for the Euler angle representation for the same rotation, there are 12 ways to project the quaternion back to Euler angles. Equation (32-34) demonstrates the conversion to Z-Y-X rotational set. In addition, the limitations in equation (11) for Euler angles representation to maintain its uniqueness still apply in this calculation.

$$q_0 = cos\left(\frac{\rho}{2}\right)cos\left(\frac{\varphi}{2}\right)cos\left(\frac{\theta}{2}\right) + sin\left(\frac{\rho}{2}\right)sin\left(\frac{\varphi}{2}\right)sin\left(\frac{\theta}{2}\right) \quad (28)$$

$$q_1 = -cos\left(\frac{\rho}{2}\right)sin\left(\frac{\varphi}{2}\right)sin\left(\frac{\theta}{2}\right) + cos\left(\frac{\rho}{2}\right)cos\left(\frac{\varphi}{2}\right)sin\left(\frac{\theta}{2}\right) \quad (29)$$

$$q_2 = cos\left(\frac{\rho}{2}\right)cos\left(\frac{\varphi}{2}\right)sin\left(\frac{\theta}{2}\right) + sin\left(\frac{\rho}{2}\right)cos\left(\frac{\varphi}{2}\right)sin\left(\frac{\theta}{2}\right) \quad (30)$$

$$q_3 = cos\left(\frac{\rho}{2}\right)cos\left(\frac{\varphi}{2}\right)sin\left(\frac{\theta}{2}\right) - sin\left(\frac{\rho}{2}\right)cos\left(\frac{\varphi}{2}\right)sin\left(\frac{\theta}{2}\right) \quad (31)$$

$$\rho = \text{atan}\left(\frac{2(q_2q_3 + q_0q_1)}{1 - 2(q_1^2 + q_2^2)}\right) \quad (32)$$

$$\varphi = \text{asin}(2(q_1q_3 - q_0q_2)) \quad (33)$$

$$\theta = \text{atan}\left(\frac{2(q_1q_2 + q_0q_3)}{1 - 2(q_2^2 + q_3^2)}\right) \quad (34)$$

2.1.6 Cayley-Klein Parameters

Cayley Klevin Parameters are a set of four complex parameters that is closely related to quaternion. The parameterization of Cayley Klein Parameters is as followed.

$$\mathrm{C}_\alpha = q_0 + \boldsymbol{i}q_3 \quad (35)$$

$$\mathrm{C}_\beta = q_2 + \boldsymbol{i}q_1 \quad (36)$$

$$\mathrm{C}_\gamma = -q_2 + \boldsymbol{i}q_1 \quad (37)$$

$$\mathrm{C}_\delta = q_4 - \boldsymbol{i}q_3 \quad (38)$$

where $\boldsymbol{i} = \sqrt{-1}$.

The Cayley-Klein Parameters follow the constraints in equations (39-41), and the rotation matrix is shown in (42).

$$C_\alpha C_\alpha{}^* + C_\gamma C_\gamma{}^* = C_\alpha C_\alpha{}^* + C_\beta C_\beta{}^* = C_\alpha C_\delta + C_\beta C_\gamma = 1 \quad (39)$$

$$C_\alpha C_\delta{}^* + C_\gamma C_\delta{}^* = 0 \quad (40)$$

$$C_\delta = C_\alpha{}^*, \qquad C_\beta = C_\gamma{}^* \quad (41)$$

Where $C_\alpha{}^*$ is the complex conjugate of C_α.

$$R_C = \begin{bmatrix} \frac{1}{2}\left(C_\alpha{}^2 - C_\beta{}^2 - C_\gamma{}^2 + C_\delta{}^2\right) & \frac{i}{2}\left(-C_\alpha{}^2 - C_\beta{}^2 + C_\gamma{}^2 + C_\delta{}^2\right) & (C_\gamma C_\delta - C_\alpha C_\beta) \\ \frac{i}{2}\left(C_\alpha{}^2 - C_\beta{}^2 + C_\gamma{}^2 - C_\delta{}^2\right) & \frac{1}{2}\left(C_\alpha{}^2 + C_\beta{}^2 + C_\gamma{}^2 + C_\delta{}^2\right) & -\boldsymbol{i}(C_\alpha C_\beta + C_\gamma C_\delta) \\ (C_\beta C_\delta - C_\alpha C_\gamma) & \boldsymbol{i}(C_\alpha C_\gamma + C_\beta C_\delta) & (C_\alpha C_\delta + C_\gamma C_\beta) \end{bmatrix} \quad (42)$$

2.1.7 Rodrigues Parameters

The Rodrigues parameters are a representation that reduces the element in the Euler Rodrigues symmetric parameters. It is defined as,

$$\begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \end{bmatrix} = \begin{bmatrix} \hat{\eta}_1 \tan(\frac{\alpha}{2}) \\ \hat{\eta}_2 \tan(\frac{\alpha}{2}) \\ \hat{\eta}_3 \tan(\frac{\alpha}{2}) \end{bmatrix} = \begin{bmatrix} \eta_1 / \ \eta_4 \\ \eta_2 / \ \eta_4 \\ \eta_3 / \ \eta_4 \end{bmatrix} \quad (43)$$

The rotation transformation between two Rodrigues parameters is shown below.

$$R_\gamma R_\tau = (\gamma, \tau) = \frac{\gamma + \tau - \gamma \times \tau}{1 - \gamma \cdot \tau} \quad (44)$$

The primarily disadvantage for Rodrigues Parameter is that there are two singularity points occur as α approaches $\pm\pi$.

2.1.8 Modified Rodrigues Parameters

The modified Rodrigues parameters are very similar to the Rodrigues parameters. It is defined as followed.

$$\begin{bmatrix}\sigma_1\\ \sigma_2\\ \sigma_3\end{bmatrix}=\begin{bmatrix}\hat{\eta}_1\tan(\frac{\alpha}{4})\\ \hat{\eta}_2\tan(\frac{\alpha}{4})\\ \hat{\eta}_3\tan(\frac{\alpha}{4})\end{bmatrix}=\begin{bmatrix}\eta_1/(1+\eta_4)\\ \eta_2/(1+\eta_4)\\ \eta_3/(1+\eta_4)\end{bmatrix} \tag{45}$$

The rotation matrix for modified Rodrigues Parameters is shown in equation (46), and the rotation transformation between two modified Rodrigues Parameters is shown in equation (47).

$$R_\sigma=\frac{1}{(1+\sigma^T\sigma)^2}\begin{bmatrix}4({\sigma_1}^2-{\sigma_2}^2-{\sigma_3}^2)-\mathrm{E}^2 & 8\sigma_1\sigma_2+4\eta_3\mathrm{E} & 8\sigma_1\sigma_3+4\eta_2\mathrm{E}\\ 8\sigma_2\sigma_1-4\eta_3\mathrm{E} & 4(-{\sigma_1}^2+{\sigma_2}^2-{\sigma_3}^2)-\mathrm{E}^2 & 2(\eta_2\eta_3+\eta_4\eta_1)\\ 8\sigma_3\sigma_1+4\eta_2\mathrm{E} & 8\sigma_3\sigma_2-4\eta_1\mathrm{E} & 4(-{\sigma_1}^2-{\sigma_2}^2+{\sigma_3}^2)-\mathrm{E}^2\end{bmatrix} \tag{46}$$

where $\mathrm{E}=1-\sigma$.

$$R_\sigma R_\Sigma=(\sigma,\Sigma)=\frac{(1-|\sigma|^2)\Sigma+(1-|\Sigma|^2)\sigma-2\Sigma\times\sigma}{1+|\sigma|^2|\Sigma|^2-2\sigma\cdot\Sigma} \tag{47}$$

Similar to Rodrigues Parameters, Modified Rodrigues Parameters experience singularity at certain attitude. However, there is only one single singularity point at $0,2\pi$ for modified Rodrigues Parameters.

2.2 Review of Attitude Estimation Algorithm

The initial design of the IMU system were using analog computer to implement simple algorithm, which are subjected to noise and calculation error. With the advance in digital computing, multiple signal processing algorithms for improving the condition of the signal were realized. However, in an inherently noisy system, it is not possible to completely eliminate the noise that contributes to the arithmetic drift. Algorithm designs for sensor fusion techniques have also played an important role in improving the accuracy of the IMU system. The following section reviews the principle of various algorithmic designs for attitude estimation.

2.2.1 Recursive Bayesian Estimation

The principle of attitude estimation algorithm is to predict and correct the input data to generate meaningful outputs in a recursive manner. Recursive Bayesian estimation is a statistical estimation technique based on the prediction and correction principles. The algorithm predicts the future states of the system based on data in the current interval and the *priori* knowledge of the system's statistical model. The observation states computed from the measurements collected at the current interval is compared with the prediction made in the previous interval. This knowledge is then used to update and correct the statistical properties of the system model. There are many implementations based on the core principles of recursive Bayesian estimation. The sections below illustrated some of the most notable designs.

2.2.2 Principle of Kalman Filter

In 1951, Rudolf Kalman introduced a recursive estimation algorithm for discrete linear filtering problem, which is now known as Kalman filter [55]. Kalman filter is a state space modeling method for a dynamic linear system with input measurements polluted by random noise. The filter is to model such conditions, produces predictions, computes the uncertainty of each sensor inputs, and determined the most likely outcome of the system. As a recursive estimator, Kalman filter only requires the measurements from the previous state to calculate the prediction and the measurements from the current state to calculate the errors and adjustments. It does not require any other memories of the measurements.

Consider a discrete linear dynamic model, the transition between the current state at time k and the previous state at time (k-1) can be described with the following model.

$$x_k = Ax_{k-1} + Bu_{k-1} + w_{k-1} \tag{48}$$

where x_k is the state vector of the current state, x_{k-1} is the state vector from the previous state, A is the transitional matrix model to transform the previous state into the current state, B is the matrix model for controlled input u_{k-1} from the previous state, and $w_{k-1}$is the process noise, which is independent and normally distributed around zero means with process noise covariance matrix Q.

$$p(w) = N(0, Q) \tag{49}$$

The model that relates the measurement to the state vector x_k of the system at time k is,

$$z_k = Hx_k + h_k \tag{50}$$

where z_k is the measurements vector at time k, H is the matrix of measurement equations that relates the state x_k to z_k, and v_k is the measurement noise at time k, which is independent with zero means normal probability with measurement noise covariance matrix R.

$$p(v) = N(0, R) \tag{51}$$

As the same vector at different instances is needed for the calculation, the following parameters are defined. $\hat{x}_k^-$ is defined as the *priori* estimation of the state at step k with the knowledge of the process prior to step k. $\hat{x}_k$ is defined as the *posteriori* estimation of the state at step k given the measurement z_k. The errors for the *priori* and *posteriori* estimations are defined as

$$e_k^- \equiv x_k - \hat{x}_k^- \tag{52}$$

$$e_k \equiv x_k - \hat{x}_k \tag{53}$$

The error covariance matrixes for the *priori* and *posteriori* estimations are,

$$P_k^- = E[e_k^- e_k^{-T}] \tag{54}$$

$$P_k = E[e_k e_k^T] \tag{55}$$

The essence of the Kalman filter is to determine the differences between the predictions and the measurements, and adjust the filter accordingly. The equation below is known as the innovation matrix, which is the difference between the *priori* prediction $H\hat{x}_k^-$, and the measurement z_k.

$$\tilde{y}_k = z_k - H\hat{x}_k^- \tag{56}$$

where $\tilde{y}_k$ is the innovation matrix.

The innovation error covariance matrix, S_k, determines residuals error between $H\hat{x}_k^-$ and z_k as shown in equation below.

$$S_k = HP_k^- H^T + R \tag{57}$$

The *posteriori* state estimate $\hat{x}_k$, is then a linear combination of the *priori* estimate $\hat{x}_k^-$, and a weighted innovation adjustment.

$$\hat{x}_k = \hat{x}_k^- + K_k\tilde{y}_k \tag{58}$$

where K_k is the optimal Kalman gain.

The Kalman gain is determined by minimizing the *posteriori* estimate covariance matrix, which can be computed as follow. The *posteriori* error covariance P_k is given as

$$P_k = cov(x_k - \hat{x}_k) \tag{59}$$

By expanding the formula above with the equations from the measurement model, innovation, and the *posteriori* state estimate.

$$P_k = cov\left(x_k - \left(\hat{x}_k^- + K_k\left(Hx_k + v_k - H\hat{x}_k^-\right)\right)\right) \tag{60}$$

As the *priori* estimate is an invariant; its process noise is not correlated with the other parameters, and has zero means normal probability with process noise covariance matrix R, the equation becomes

$$\begin{aligned} P_k &= (I - K_k H_k)\bar{P_k}(I - K_k H_k)^T + K_k R_k K_k^T \\ &= \bar{P_k} - K_k H_k \bar{P_k} - \bar{P_k} H_k^T K_k^T + K_k S_k K_k^T \end{aligned} \tag{61}$$

The optimal Kalman gain minimizes the *posteriori* error covariance estimate to zero. By putting the derivative of the *posteriori* estimate P_k with respect to K_k equals to zero, the optimal Kalman gain can then be determined.

$$\frac{\partial P_k}{K_k} = 0 = 2(-\left(H_k \bar{P_k}\right)^T + K_k S_k) \tag{62}$$

$$K_k = \bar{P_k} H_k^T S_k^{-1} \tag{63}$$

Kalman filter can be separated into 2 major sets of equations, which are the time updates equations and the measurements update equations. The time update equations predict the *priori* estimates at time k with the knowledge of the current states and error covariance at time k-1 in equation (64) respectively.

$$\bar{\hat{x}_k} = A\hat{x}_{k-1} + Bu_{k-1} \tag{64}$$

$$\bar{P_k} = AP_{k-1}A^T + Q \tag{65}$$

The measurements update equations use the measurements acquired with the *priori* estimates to calculate the *posteriori* estimates.

$$S_k = H\bar{P_k} H^T + R \tag{66}$$

$$K_k = \bar{P_k} H_k^T S_k^{-1} \tag{67}$$

$$\hat{x}_k = \bar{\hat{x}_k} + K_k \tilde{y}_k, \qquad \tilde{y}_k = z_k - H\bar{\hat{x}_k} \tag{68}$$

$$P_k = (I - K_k H_k)\bar{P_k} \tag{69}$$

The *posteriori* estimate is then use to predict *priori* estimate at the next time step. As displayed from the equations above, no further information is required beside the state and error covariance from the previous state. The algorithm is extremely efficient and suitable for the tracking problem where multiple concurrent input measurements are required.

2.2.3 Principle of Extended Kalman Filter

Kalman filter makes an assumption regarding to the linearity of the dynamic system. Extended Kalman Filter (EKF) is an extension of the Kalman filter, developed to tackle the non-linear problem by approximating the non-linear system by linearization. [56]. The basic concept is to determine the estimate through linearization of current estimates by calculating the Jacobian of the process and measurement matrixes. The process and measurement models are redefined as followed,

$$x_k = f(x_{k-1},\ u_{k-1}, w_{k-1}) \tag{70}$$

$$z_k = h(x_k,\ v_k) \tag{71}$$

where f is the non-linear function to transform the previous state into the current state, and h is the non-linear function that relates the state x_k to z_k. The non-linear process and measurement model can be expressed as,

$$x_k \approx \tilde{x}_k\ + A_{k-1}(x_{k-1} - \hat{x}_{k-1}) + W_{k-1}w_{k-1} \tag{72}$$

$$z_k \approx \tilde{z}_k\ + H_k(x_k - \hat{x}_k) + V_k v_k \tag{73}$$

where $\tilde{x}_k$ is the approximate process state vector, and $\tilde{z}_k$ is the approximate measurement vector.

A_{k-1} is the Jacobian matrix of partial derivate of f with respect to x.

$$A_{k-1_{[i,j]}} = \frac{\partial f_{[i]}}{\partial x_{[j]}}(\hat{x}_{k-1},\ u_{k-1}) \tag{74}$$

W_{k-1} is the Jacobian matrix of partial derivate of f with respect to w.

$$W_{k-1_{[i,j]}} = \frac{\partial f_{[i]}}{\partial w_{[j]}}(\hat{x}_{k-1},\, u_{k-1}) \tag{75}$$

H_k is the Jacobian matrix of partial derivate of h with respect to x.

$$H_{k_{[i,j]}} = \frac{\partial h_{[i]}}{\partial x_{[j]}}(\tilde{x}_k\,) \tag{76}$$

V_k is the Jacobian matrix of partial derivate of h with respect to w.

$$V_{k_{[i,j]}} = \frac{\partial h_{[i]}}{\partial w_{[j]}}(\tilde{x}_k\,) \tag{77}$$

The error for *priori* and *posteriori* estimations are redefined as

$$\tilde{e}_{x_k} \approx A_{k-1}(x_{k-1} - \hat{x}_{k-1}) + \varepsilon_k \tag{78}$$

$$\tilde{e}_{z_k} \approx H_k \tilde{e}_{x_k} + \eta_k \tag{79}$$

where ε_k is defined as an independent random variable with zero means and covariance matrix $W_{k-1}QW_{k-1}{}^T$,

$$p(\varepsilon_k) \sim N(0, W_{k-1}QW_{k-1}{}^T) \tag{80}$$

and $\eta_k$is defined as an independent random variable with zero means and covariance matrix $V_kRV_k{}^T$.

$$p(\eta_k) \sim N(0, V_kRV_k{}^T) \tag{81}$$

The time updates equations for the extended Kalman filter are updated to

$$\bar{\hat{x}_k} = f(x_{k-1},\, u_{k-1}) \tag{82}$$

$$\bar{P_k} = A_kP_{k-1}A_k{}^T + W_kQ_{k-1}W_k{}^T \tag{83}$$

And the measurement update equations are

$$S_k = H_k\bar{P_k}H_k{}^T + V_kRV_k{}^T \tag{84}$$

$$K_k = \bar{P_k}H_k^T S_k^{-1} \tag{85}$$

$$\hat{x}_k = \bar{\hat{x}_k} + K_k\tilde{y}_k, \qquad \tilde{y}_k = z_k - h(\bar{\hat{x}_k}) \tag{86}$$

$$P_k = (I - K_k H_k)\bar{P_k} \tag{87}$$

The primarily difference between KF and EKF is the linearization of the system state by the calculation of the Jacobian matrixes. The prediction and update equations are essentially the same. It should be noted that this is a first order EKF, where an assumption that the effects of higher order terms are negligible. However, in some cases, where higher order terms affect the accuracy of the filter, Hessian matrixes will be required in the calculation. In addition, Kalman filter is considered as an optimal filter with sub-optimal conditions owning the assumptions of the system states. Extended Kalman filter becomes a non-optimal filter due to the linearization process. The system models and noise processes must be designed carefully to ensure stability of the algorithm and prevent it diverging.

2.2.4 Kalman and Extended Kalman filter with QUEST

There are numerous variations of the KF and EKF based on the interpretation in optimizing the error estimation. One of most notable families is using the QUaternion ESTimator (QUEST) algorithm. The QUEST family tackles the minimization method of the Wahba's problem for quaternion estimation [57]. Consider the following system,

$$v_{k_A} = R^{AB} v_{k_B} \;\; k = 1{:}\, N \tag{88}$$

where v_{k_A} is a set of N vectors representing the measurements from sensor A, v_{k_B} is a set of N vectors representing the measurement from sensor B, and R^{AB} is a rotation matrix to transform v_{k_B} to v_{k_A}. The goal is to obtain the R^{AB} that minimizes the error. This is done by computing the R^{AB} that minimizes the lose function,

$$J(R^{AB}) = 0.5 \sum_{k=1}^{N} w_k \left| v_{k_A} - R^{AB} v_{k_B} \right|^2 \tag{89}$$

where J is the lose function, w_k is non-negative weights, and N is total number of measurements.

There are many approaches to solve this problem. In 1978, Lerner proposed the q-method [58]. By expanding the loss function,

$$J(R^{AB}) = 0.5\sum_{k=1}^{N} w_k \left(v_{k_A} - R^{AB} v_{k_B}\right)^T \left(v_{k_A} - R^{AB} v_{k_B}\right)$$
$$=0.5\sum_{k=1}^{N} w_k \left({v_{k_A}}^T v_{k_A} + {v_{k_B}}^T v_{k_B} - 2{v_{k_A}}^T R^{AB} v_{k_B}\right),\ where\ {v_{k_A}}^T v_{k_A} = {v_{k_B}}^T v_{k_B} = 1 \quad (90)$$
$$=\sum_{k=1}^{N} w_k \left(1 - {v_{k_A}}^T R^{AB} v_{k_B}\right)$$

According to equation (90), minimizing J is to maximize the gain function defined as,

$$g(R^{AB}) = \sum_{k=1}^{N} w_k \, {v_{k_A}}^T R^{AB} v_{k_B} \quad (91)$$

For quaternion, the gain function can be defined as,

$$g(q) = q^T K q \quad (92)$$

where K is a symmetric traceless matrix

$$K \equiv \begin{bmatrix} B + B^T - tr[B]\mathbf{I}_{\times} & Z \\ Z^T & tr[B] \end{bmatrix} \quad (93)$$

$$\mathrm{B} = \sum_{k=1}^{N} w_k \, v_{k_A} {v_{k_B}}^T \quad (94)$$

$$Z = [B_{23} - B_{32}\ B_{31} - B_{13}\ \ B_{12} - B_{21}]^T \quad (95)$$

Adding the Lagrange multiplier (96) and differentiate the gain function (97),

$$g(q) = q^T K q - \lambda q^T q \quad (96)$$

$$Kq = \lambda q \quad (97)$$

The optimal attitude is the largest eigenvalue of Kthat maximizes the gain function. This method obtains the least-square optimal estimate of the orientation by solving the eigenvector directly. Shuster proposed the QUEST algorithm to solve the minimization problem with much higher efficient by approximating the largest eigenvalue [59].

Based on equations (90) and (91), the optimal eigenvalue can be expressed as,

$$\begin{aligned} \lambda_{opt} &= \sum_{k=1}^{N} w_k - J \\ &\approx \sum_{k=1}^{N} w_k \end{aligned} \quad (98)$$

The optimal quaternion can then be determined by first calculating the Rodrigues parameters (η), which can be converted to quaternion as shown in equation (100).

$$\eta = \frac{1}{\left[\lambda_{opt} + tr[B] - S\right]} Z \tag{99}$$

$$q = \frac{1}{\sqrt{1 + \eta^T \eta}} \begin{bmatrix} 1 \\ \eta \end{bmatrix} \tag{100}$$

Filter QUEST [60] and REQUEST [61] combines the Kalman measurement stage with the QUEST estimation procedure. The primarily differences between the two algorithm is that filter QUEST uses the attitude matrix in predicting and updating B, while REQUEST predicts and updates the Kmatrix. Optimal REQUEST [62] is similar to REQUEST but uses stochastic process noise model to solve for optimal filter gain. Extended QUEST [63] uses the linearization in the prediction stage as in the EKF.

2.2.5 Principle of Unscented Kalman Filter

One of the main issues with EKF is that the normality of the distribution of the random variables is lost during the non-linear transformation. In 1996, Julier proposed a new strategy to maintain the normality of the distribution during non-linear transformation [64]. The principle is to select a minimal sets of chosen sample points to capture the true mean and covariance of the Gaussian random variables, and propagate through the non-linear system. This technique is known as unscented transformation. The following algorithm lay out the basic for unscented transforms.

Assuming a random variable x with dimension L, which has known mean $\bar{x}$ and covariance P_x, undergoes a non-linear function f.

$$y = f(x)) \tag{101}$$

The statistic of y can be computed by defining a matrix χ of $2L + 1$ sample vectors χ_i.

$$\chi_0 = \bar{x} \tag{102}$$

$$\lambda = \alpha^2(L + \kappa) - L \tag{103}$$

$$\chi_i = \bar{x} + \left(\sqrt{(L + \lambda)\, P_x}\right)_i \quad ; \quad i = 1, \dots, L \tag{104}$$

$$\chi_i = \bar{x} - \left(\sqrt{(L + \lambda)\, P_x}\right)_{i-L} \quad ; \quad i = L + 1, \dots, 2L \tag{105}$$

where α is a constant that decides the spread of sample points around $\bar{x}$, and κ is a scaling parameter. The weights $\mathcal{W}_i$ of the sample vectors are defined with the following equations.

$$\mathcal{W}_0^{(m)} = \frac{\lambda}{(L + \lambda)} \tag{106}$$

$$\mathcal{W}_0^{(c)} = \frac{\lambda}{(L + \lambda)} + (1 - \alpha^2 + \beta) \tag{107}$$

$$\mathcal{W}_i^{(m)} = \mathcal{W}_i^{(c)} = \frac{1}{2(L + \lambda)} \quad ; \quad i = 0, \dots, 2L \tag{108}$$

where β represents the *priori* information of the distribution of the random variable x.

The non-linear function is then applied to the sample vectors.

$$\mathcal{Y}_i = f(\chi_i) \quad ; \quad i = 0, \dots, 2L \tag{109}$$

The mean and covariance of y can then be calculated with

$$\bar{y} \approx \sum_{i=0}^{2L} \mathcal{W}_i^{(m)}\, \mathcal{Y}_i \tag{110}$$

$$P_y \approx \sum_{i=0}^{2L} \mathcal{W}_i^{(c)}\, (\mathcal{Y}_i - \bar{y})(\mathcal{Y}_i - \bar{y})^T \tag{111}$$

Unscented Kalman filter (UKF) is an adaptation of the unscented transform algorithm into the Extended Kalman filter. Using the non-linear model from previous section, a random vector x_o is defined as the initial state of the vector with known mean and covariance.

$$\mu_0 = E[x_o] \tag{112}$$

$$P = E[(x_o - \mu_0)(x_o - \mu_0)^T] \tag{113}$$

In the time update steps, an augmented state is introduced, which consists of both the original state and the process noise.

$$x_{k-1}^{a} = [x_{k-1}^{T}\ w_{k-1}^{T}]^{T} \tag{114}$$

The covariance matrix x_k^a is

$$P_{k-1}^{a} = \begin{bmatrix} P_{k-1} & 0 \\ 0 & Q \end{bmatrix} \tag{115}$$

Following the unscented transform algorithm by calculating the sample points,

$$\chi_{k-1}^{0} = x_{k-1}^{a} \tag{116}$$

$$\chi_{k-1}^{i} = x_{k-1}^{a} + \left(\sqrt{(L+\lambda)P_{k-1}^{a}}\right)_{i} \quad ; \quad i = 1, \dots, L \tag{117}$$

$$\chi_{k-1}^{i} = x_{k-1}^{a} - \left(\sqrt{(L+\lambda)P_{k-1}^{a}}\right)_{i} \quad ; \quad i = L+1, \dots, 2L \tag{118}$$

The sample points are then propagated to the non-linear transitional function,

$$\chi_{k}^{x} = f[\chi_{k-1}^{x},\ u_{k-1}, \chi_{k-1}^{w}] \tag{119}$$

The mean of covariance of the sample points are,

$$\bar{\hat{x}_k} = \sum_{i=0}^{2L} \mathcal{W}_i^{(m)}\, \chi_{i\ k-1}^{x} \tag{120}$$

$$P_k = \sum_{i=0}^{2L} \mathcal{W}_i^{(c)}\, [\chi_{i\ k-1}^{x} - \bar{\hat{x}_{k-1}}][\chi_{i\ k-1}^{x} - \bar{\hat{x}_{k-1}}]^{T} \tag{121}$$

In the measurement model, the augmented state is defined as,

$$x_{k}^{a} = [x_{k}^{T}\ v_{k}^{T}]^{T} \tag{122}$$

The covariance matrix of x_k^a is

$$P_{k}^{a} = \begin{bmatrix} P_k & 0 \\ 0 & R \end{bmatrix} \tag{123}$$

Using unscented transform yields,

$$\chi_{k}^{0} = x_{k}^{a} \tag{124}$$

$$\chi_{k}^{i} = x_{k}^{a} + \left(\sqrt{(L+\lambda)P_{k}^{a}}\right)_{i} \quad ; \quad i = 1, \dots, L \tag{125}$$

$$\chi_{k}^{i} = x_{k}^{a} - \left(\sqrt{(L+\lambda)P_{k}^{a}}\right)_{i} \quad ; \quad i = L+1, \dots, 2L \tag{126}$$

The sample points are propagated into the non-linear measurement function,

$$\mathrm{X}_k^x = h[\chi_k^x, \chi_k^v] \tag{127}$$

The measurement vector z_k and its covariance can be expressed as,

$$\hat{z}_k^- = \sum_{i=0}^{2L} \mathcal{W}_i^{(m)} \mathrm{X}_{i\,k}^x \tag{128}$$

$$P_{\hat{z}_k\,\hat{z}_k} = \sum_{i=0}^{2L} \mathcal{W}_i^{(c)} [\mathrm{X}_{i\,k}^x - \hat{z}_k^-][\mathrm{X}_{i\,k}^x - \hat{z}_k^-]^T \tag{129}$$

The process vector and measurement vector cross covariance matrix is calculated.

$$P_{\hat{x}_k\,\hat{z}_k} = \sum_{i=0}^{2L} \mathcal{W}_i^{(c)} [\chi_{i\,k-1}^x - \hat{x}_{k-1}^-][\mathrm{X}_{i\,k}^x - \hat{z}_k^-]^T \tag{130}$$

The Kalman gain is determined by both the error covariance matrix of measurement vector and the error covariance of between the process and measurement vectors.

$$K_k = P_{\hat{x}_k\,\hat{z}_k} P_{\hat{z}_k\,\hat{z}_k}^{-1} \tag{131}$$

Posteriori estimates and the covariance matrix can then be evaluated.

$$\hat{x}_k = \hat{x}_k^- + K_k\tilde{y}_k, \quad \tilde{y}_k = z_k - \hat{z}_k^- \tag{132}$$

$$P_k = P_k^- - K_k P_{\hat{z}_k\,\hat{z}_k} K_k^T \tag{133}$$

One of the main advantages of unscented Kalman filter over extended Kalman filter is that it does not require the derivation and computation of a Jacobian or Hessian matrixes for the system model, which can be problematic for complex system. Similar to other Kalman filters, unscented Kalman filter operates under the assumption that the system models are Gaussian in nature. Hence, non-Gaussian system generally yields inferior result with these filters.

2.2.6 Sequential Monte Carlo Methods

Consider a discrete non-linear non-Gaussian stochastic system that has the following process and measurement models.

$$x_k = f_k(x_{k-1}, w_{k-1}) \tag{134}$$

$$z_k = h_k(x_k,\ v_k) \tag{135}$$

where f_k is a function of unknown properties that ties the previous state and the current state, and h_k is the a function with unknown properties that links the state x_k to z_k.

As the statistical properties of these processes are unknown, the Bayesian approach to this problem is to construct a posterior probability density function of the predicting estimate given all previous observations,

$$p(x_k|z_{1:k}),\ z_{1:k} \triangleq \{z_1, z_2 \dots z_k\} \tag{136}$$

The prior pdf at time k can be expressed by the Chapman–Kolmogorov equation,

$$p(x_k|z_{1:k-1}), = \int p(x_k|x_{k-1}, z_{1:k-1})p(x_{k-1}|z_{1:k-1})\, dx_{k-1} \tag{137}$$

where $p(x_k|x_{k-1}, z_{1:k-1})$ is the predictive conditional density of the process model, and $p(x_{k-1}|z_{1:k-1})$ is the posterior pdf from previous interval.

The posterior pdf at time k is determined by,

$$p(x_k|z_{1:k}) = \frac{p(z_k|x_k)p(x_k|z_{1:k-1})}{p(z_k|z_{k-1})} \tag{138}$$

where

$$p(z_k|z_{k-1}) = \int p(z_k|x_k)p(x_k|z_{1:k-1})\, dx_k \tag{139}$$

In equation (138), $p(z_k|x_k)$ is the likelihood function described by the measurement model, and $p(z_k|z_{k-1})$ is the normalizing constant. Equation (137) is regarded as the prediction stage of the estimation algorithm, while equation (138) is the update stage. This recursion forms the bases of the recursive estimation algorithm. However, the posterior density is an intractable inference problem that cannot be determined analytically as the size of the dataset is sequentially expanding.

Sequential Monte Carlo (SMC) method [65], or particle filter (PF), is a technique to tackle the intractable integral in the posterior density approximation of the sequential Bayesian estimation with the Monte Carlo method [66,67]. Particle filter can be considered a brute force approach to approximate the posterior density with a large sum of independent and identically distributed random variables or particles from the same probability density space.

Consider a set of *N* independent random samples are drawn from a probability density $p(x_k|z_k)$,

$$x_k(i) \sim p(x_k|z_{1:k}), \qquad i = 1:N \tag{140}$$

The Monte Carlo representation of the probability density can then be approximated as,

$$p(x_k|z_{1:k}) \approx \frac{1}{N}\sum_{i=1}^{N} \delta_{x_k(i)}(x_k) \tag{141}$$

where $\delta_{x(i)}$ is the Dirac delta function of the points mass. Using this interpretation, the expectation of the any testing function $h(x)$ is given by

$$\begin{aligned} \mathbb{E}\big(h(x_k)\big) = \int h(x_k)p(x_k|z_{1:k})dx_k \;&\approx \int h(x_k)\frac{1}{N}\sum_{i=1}^{N}\delta_{x_k(i)}(x_k)\,dx_k \\ &= \frac{1}{N}\sum_{i=1}^{N} h(x_k(i)), \quad i = 1:N \end{aligned} \tag{142}$$

In practice, sampling from $p(x)$ directly is usually not possible due to latent hidden variables in the estimation. Alternatively, samples are drawn from a different probability density $q(x_k|z_{1:k})$ is proposed,

$$x_k(i) \sim q(x_k|z_{1:k}), \qquad i = 1:N \tag{143}$$

which is generally known as the importance function or the importance density. A correction step is then used to ensure the expectation estimation from the probability density $q(x_k|z_{1:k})$ remains valid. The correction factor, which is generally regarded as

the importance weights of the samples ($w_k(i)$), is proportional to the ratio between the target probability density and the proposed probability density,

$$w_k(i) \propto \frac{p(x_k|z_{1:k})}{q(x_k|z_{1:k})} \quad i = 1{:}N \tag{144}$$

The importance weights are normalized,

$$\sum_{i=1}^{N} w_k(i) = 1 \tag{145}$$

Based on the sample drawn from equation (143), the posterior probability density becomes,

$$p(x_k|z_{1:k}) = \frac{p(z_k|x_k|z_{k-1})p(x_k|z_{k-1})}{p(z_k|z_{k-1})} \tag{146}$$

$$= \frac{p(z_k|x_k)p(x_k|x_{k-1})}{p(z_k|z_{k-1})} p(x_k|z_{1:k-1}) \tag{147}$$

$$\propto p(z_k|x_k)p(x_k|x_{k-1})p(x_k|z_{1:k-1}) \tag{148}$$

And the importance weight from equation (144) becomes,

$$w_k(i) \propto \frac{p\big(z_k\big|x_k(i)\big)p\big(x_k(i)\big|x_{k-1}(i)\big)p(x_{1:k-1}(i)|z_{1:k-1})}{q\big(x_k(i)\big|x_{1:k-1}(i)\big)q(x_{1:k-1}(i)|z_{1:k-1})}, \quad i = 1{:}N \tag{149}$$

$$= w_{k-1}(i) \frac{p\big(z_k\big|x_k(i)\big)p\big(x_k(i)\big|x_{k-1}(i)\big)}{q\big(x_k(i)\big|x_{1:k-1}(i)\big)} \tag{150}$$

$$\propto w_{k-1}(i) \frac{p\big(z_k\big|x_k(i)\big)p\big(x_k(i)\big|x_{k-1}(i)\big)}{q\big(x_k(i)\big|x_{k-1}(i)\big)} \tag{151}$$

The posterior probability density can then be approximated empirically by,

$$p(x_k|z_{1:k}) \approx \sum_{i=1}^{N} w_k(i)\, \delta_{x_k(i)}(x_k) \tag{152}$$

The expectation of the estimation from equation (142) can be expressed as,

$$\mathbb{E}\big(h(x_k)\big) = \int h(x_k)p(x_k|z_{1:k})dx_k \approx \int h(x_k) \sum_{i=1}^{N} w_k(i)\, \delta_{x_k(i)}(x_k)$$
$$= \sum_{i=1}^{N} w_k(i)h(x_k(i)), \quad i = 1{:}N \tag{153}$$

The technique demonstrated by equations (149-152) is regarded as the sequential importance sampling (SIS) procedure. However, the issue with SIS is that the

importance weights will be concentrated on a few samples while the rest of them become negligible after a few recursions. This is known as the degeneracy problem with particle filter. A frequent approach to counter this problem is resampling the samples such that they are all equally weighted based on the posterior density. However, since resampling the samples introduces Monte Carlo error, resampling should not be performed in every recursion. It should only be executed when the distribution of the importance weight of the sample has been degraded. The state of the samples is determined by the effective sample size, which is defined by, [68]

$$N_{eff} = \frac{N}{1 + var(w_k{}^*(i))}, \quad i = 1{:}N \tag{154}$$

where $w_k{}^*(i)$ is the true weight of the sample,

$$w_k{}^*(i) = \frac{p(x_k|z_{1:k})}{q\left(x_k(i)\middle|x_{k-1}(i)\right)}, \quad i = 1{:}N \tag{155}$$

However, as the true weight of the sample cannot be determined directly, the following method is used to approximate the effective sample size empirically with the normalized weights. [69]

$$N_{eff} = \frac{1}{\sum_i^N w_i{}^2}, \quad i = 1{:}N \tag{156}$$

Resampling is performed when N_{eff} drops below a predetermined threshold N_{th}, which is done by relocating the samples with small weight to the samples with higher weights, hence, redistributing the weights of the particles.

Particle filter is a remarkably robust algorithm for non-linear non-Gaussian system estimation. The primarily drawback, is the exhaustive computation requirement, as it requires substantial amount of samples to accurately approximate and capture the statistical properties of the system

2.2.7 Principles of Complementary filter

Complementary filter is a special class of estimation filters that filters and performs estimations based on measurements from the signals that have complementary spectral properties [70,71,72]. Consider two set of noisy measurements x and y that are generated as a result of signal z, and assuming that signal x is dominated by low frequency noise and signal y is dominated by high frequency noise. The estimation of the original signal ($\hat{z}$) can be determined with the strategy in Figure 2-2.

In the case of IMU composing of magnetometers, accelerometers and rate gyroscopes sensors, the complementary filter can be modeled for a first order integrator system,

$$\dot{x} = y \tag{157}$$

where $\dot{x}$ is the orientation estimation from the gyroscopes that is dominated by high frequency noise, and y is the orientation determined from the accelerometers and magnetometers inputs, that contains primarily low frequency noise. Figure 2-3 shows the block diagram for the complementary filter for the IMU system.

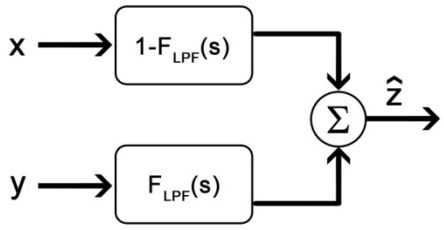

Figure 2-2 - Principle of Complementary Filter ($F_{LPF}(s)$ is a low pass filter)

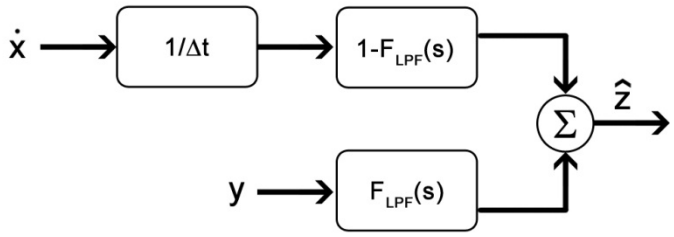

Figure 2-3 - Principle of Complementary Filter for IMU system

3. Operating Principle of Inertial Sensors

3.1 Inertial Measurement Unit Reference Axes and Frames

Tracking system based on inertial sensing technologies utilizes a different reference system comparing to the other systems that use external observer system as their point of reference. Inertial tracking systems utilize the Earth's gravitational and magnetic fields as the external frame of reference. This is generally referred as the inertial reference frame.

In order to discuss the positioning and orientations techniques, the following reference frames and axes are used as shown in Figure 3-1. Pitch, Roll and, Yaw represents the rotation of the x, y, and z axes of the object in its inertial reference frame respectively. Gravity field is used as an external reference for the pitch and row rotations of the object, while the azimuth angle, determined by the deviation between the object heading and local magnetic field is used as the reference of yaw rotation.

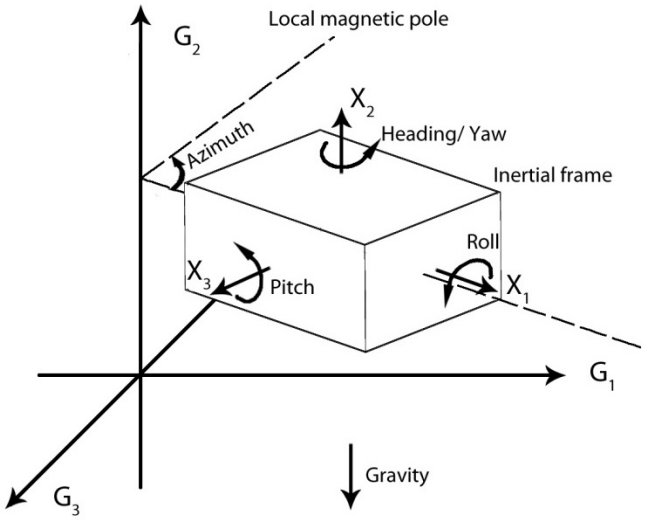

Figure 3-1 - Reference frames and axes of the IMU system

3.2 Measuring Principle of Accelerometer

Accelerometer is a device used to measure acceleration of an object in both inertial and Newtonian frames. In principle, an accelerometer is equivalent to a mass attached to a spring as shown in Figure 3-2. In Figure 3-2a, a mass *m*, is attached to a base by a spring with spring constant *k* at a relaxed length of *x0* with no external force applied to it. Newton's second law of motion states that an external force *F* is required to accelerate a mass.

$$F = ma \tag{158}$$

In Figure 3-2b, the assembly accelerates and the spring is stretched to provide enough force to accelerate the mass. Hooke's Law states that a force will be acting on the spring if it is extended from its equilibrium position Δx.

$$F = k\Delta x \tag{159}$$

By equating the Newton's second law of motion and the Hooke's law, the acceleration of a mass can be calculated from the spring constant, displacement and mass alone.

$$F = k\Delta x = ma \tag{160}$$

$$a = \frac{k\Delta x}{m} \tag{161}$$

where a is acceleration; k is the spring constant; m is the mass and Δx is the displacement.

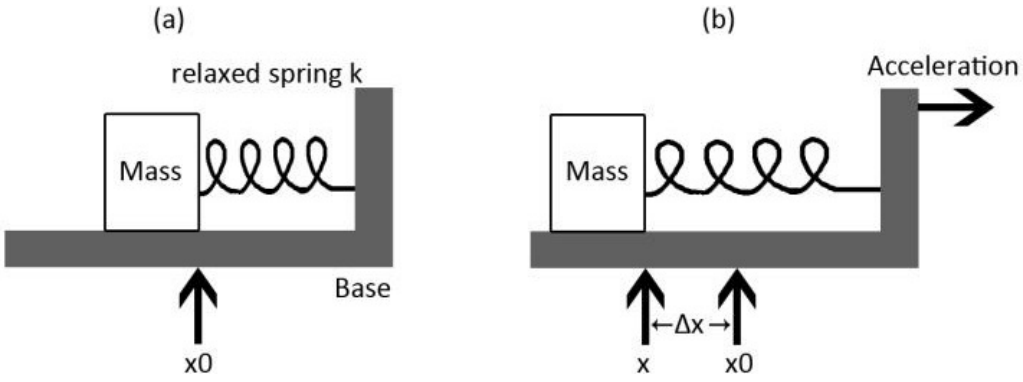

Figure 3-2 - Conceptual illustration of accelerometers

Many of the accelerometers are designed based on this principle. One of the common designs for micromachined accelerometer is an array of microcantilevers where the bending of the cantilever beam (displacement) has a relationship with acceleration. This leads to the secondary effect of accelerometers, which are vibration sensing. The transient acceleration of the spring mass model in Figure 3-2 will be as follow.

$$a(t) = -\omega^2 x_p \sin(\omega t),\ \omega = 2\pi f \tag{162}$$

and the mass motion is

$$\Delta x = \frac{m x_p}{k} \omega^2 \sin(\omega t) \tag{163}$$

where f is applied frequency, x_p is the initial peak position, t is time.

The oscillation of the cantilever beam may introduce unwanted noise depending on the application. Hence, there are many types of accelerometers with different frequencies ranges, which balance between the displacement and the damping coefficient. For the IMU system, the focus is to obtain acceleration from the motion and not the vibration. The accelerometer should have a higher sensitivity on Δx.

Using the equation of motion, the basic calculation for position from the data from accelerometer is to integrate acceleration over time twice as shown below,

$$v = \int a\Delta t = v_i + a\Delta t \tag{164}$$

$$s = \int v\Delta t = s_i + v_i\Delta t + \frac{1}{2}a\Delta t^2 \tag{165}$$

where a is acceleration, v is velocity, v_i is velocity of the previous state, s is position, s_i is position from the previous state, and Δt is time interval.

Upon close examination, one will notice that the velocity and position from the previous states also contributes the calculation of the current states. In other words, if there is any noise and error from the previous states, it will be accumulated. This is known as the arithmetic drift error. The most difficult part of designing the IMU system is the ability to control and minimize this drift, which was discussed previously in Chapter 2.

Besides of positioning, certain accelerometer is also capable of measuring orientations. Gravity is a unique external reference vector for certain type of accelerometer. In Figure 3-3, the illustration shows the effect of gravity to the microcantilever. In Figure 3-3a, when it is placed perpendicular to the gravity vector, the beam bends towards gravity, indicating 1g of acceleration towards the Earth. As the microcantilever is tilted towards the gravity vector as in Figure 3-3b, the effect of gravity lessens, and it decreases to less than 1g of acceleration. When the gravity is completely parallel to gravity as shown in Figure 3-3c, there is no deflation on the microcantilever from gravity. As a result, 0g of acceleration would be observed. According to this principle, tilting of the accelerometer can be observed and calculated

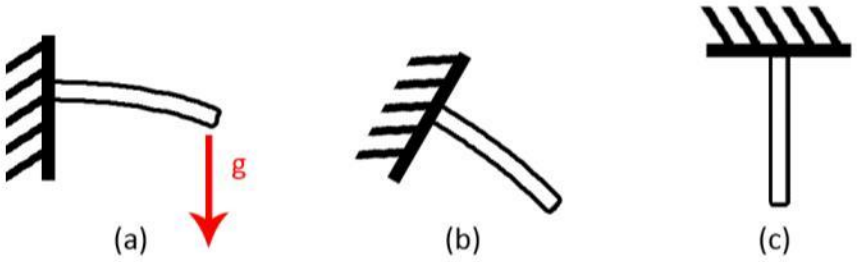

Figure 3-3 - Illustration of the micro-cantilever to the effect of gravity. Gravity has full effect on the microcantilever in a, and no effect in c.

In order to calculate the 3D tilting of the object, a tri-axial accelerometer is needed. Using simple trigonometry, the tilt angles on an object can be calculated.

$$\rho_a = atan\left(\frac{Ax}{\sqrt{A_y^2 + A_z^2}}\right) \tag{166}$$

$$\varphi_a = atan\left(\frac{Ay}{\sqrt{A_x^2 + A_z^2}}\right) \tag{167}$$

$$\theta_a = atan\left(\frac{Az}{\sqrt{A_x^2 + A_y^2}}\right) \tag{168}$$

where Ax , Ay , Az are the acceleration vectors at x, y, or z axis, and ρ_a, φ_a , and θ_a are tilt angles relative to x, y and z-axis respectively.

While the accelerometer is capable of measuring tilting relative to gravity, any orientation changes on the plane that is perpendicular to gravity will have no effect on the output of the sensors. Additionally, the root sum square in equations (166-168) would yield the same result for the unit in its original and its mirror orientations on the diagonally opposite quadrants. This leads the introduction to the second type of inertial sensors that are commonly used in IMU system.

3.3 Measuring Principle of Gyroscopes

Gyroscope is an instrument that measures angular motion of the object within the inertial frame. The classical illustration of the principle of gyroscope is shown in Figure 3-4. It consists of a motor with mass, which is suspended on three frictionless supporting rings called gimbals. Based on the conservation of angular momentum, as the rotor spins, the angular momentum of the rotor stabilizes the gyroscope and maintains constant positions with respect to horizon or direction.

Commercially available gyroscopes are generally classified into 2 different categories, which are mechanical based and optical based. This refers to the physical operating principle of the sensors. The primary difference between the two is that optical gyroscope has no moving parts as compare to the mechanical counterpart. Ring laser gyroscope (RLG) is one of the most accurate designs available in the market, which has less than 0.01^{o}/hour error. **[73]** However, due to its size and weight, it is not suitable to mount it onto human for motion tracking.

The most suitable type of gyroscope for motion tracking is the micro-machined vibratory based gyroscope. It is very small and has a low power requirement, which is ideal for wireless tracking application. Figure 3-5 illustrated the working principle of a vibratory gyroscope. The proof mass m is driven to oscillate at predetermined amplitude. When the proof mass is being rotated around an axis, the Coriolis force causes the proof mass to move in a different mode of oscillation. The Coriolis force Fc is calculated by the following equation.

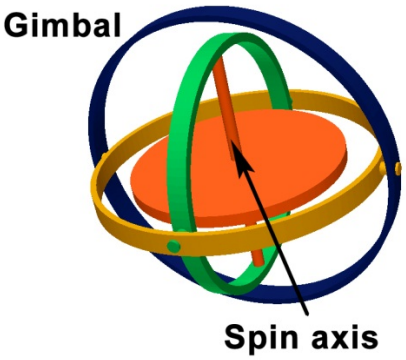

Figure 3-4 - Principle of gyroscope

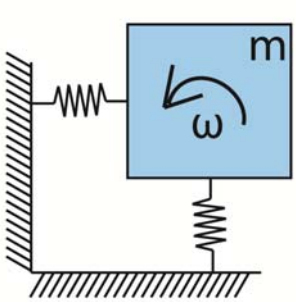

Figure 3-5 - Operating principle of vibratory gyroscope

$$F_c = -2m(v \times \omega) \tag{169}$$

where v and ω are the velocity and angular velocity of the proof mass respectively.

Using this relationship, the angular velocity of the motion can then be determined by monitoring the shift in oscillation of the proof mass.

Similar to the accelerometer, the relative orientation of the object can be calculated by the equation of motion.

$$\angle = \int \omega \Delta t = \angle_i + \omega \Delta t \tag{170}$$

where $\angle$ is the angle of orientation, and $\angle_i$ is the orientation from previous state.

Due to the integration, the orientation calculation experiences the same numerical drifting problem as stated earlier. One of the most common methods to reduce this drift is to compare the result with the orientation calculated from accelerometers with a sensor fusion technique. Since the calculation for orientation based on gravity does not involve any numerical integration, the output is very stable and drift is minimal. However, as mentioned previously, the accelerometer is incapable

of resolving orientation changes when the rotation is perpendicular to the gravity vector. Hence, it brings in an addition component to the IMU system.

3.4 Measuring Principle of Magnetometers

The most convenient way to measure heading or azimuth is by using a compass and calculate the relative change in angle compare to the magnetic north. Magnetometer is an instrument to measure the orientation difference between the sensor and the magnetic field of the earth. The principle is that the magnetic flux passing through a coil depends on its orientation with respect to the magnetic field lines of the Earth as shown in Figure 3-6.

The magnetometer determines the heading direction angle (azimuth) of the unit. On a flat 2D plane (XY) that is parallel to the Earth's surface, the calculation of the azimuth is shown below.

$$azimuth = \tan^{-1}\frac{M_y}{M_x} \tag{171}$$

$$azimuth = \frac{\pi}{2}\ (x = 0, y < 0) \tag{172}$$

$$azimuth = \frac{3\pi}{4}\ (x = 0, y > 0) \tag{173}$$

$$azimuth = \pi - (\tan^{-1}\frac{M_y}{M_x})\ (x < 0) \tag{174}$$

$$azimuth = -\left(\tan^{-1}\frac{M_y}{M_x}\right)(x > 0, y < 0) \tag{175}$$

$$azimuth = 2\pi - (\tan^{-1}\frac{M_y}{M_x})\ (x > 0, y > 0) \tag{176}$$

where M_x is the magnetic disturbance experienced on the x-axis, M_y is the magnetic disturbance on the y –axis.

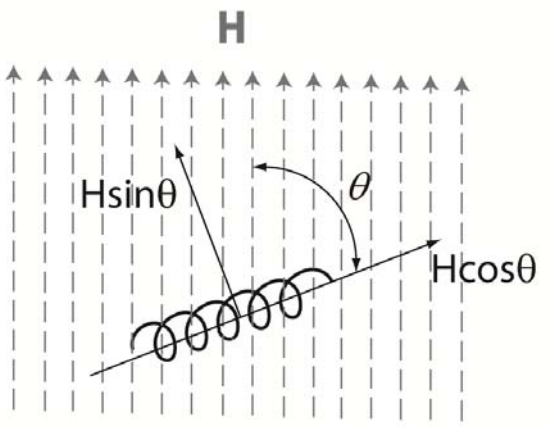

Figure 3-6 - Principle of magnetometer. The dash arrows represent the direction of the magnetic field H, and angle θ is the deviation angle between the coil and the magnetic field

However, the unit is not always going to be parallel to the Earth's surface, and there is a significant amount of error if the unit is tilted away from its optimal position. The tilt angles from the accelerometer (ρ_a, φ_a) can be used to compensate the data from the magnetometer.

$$M_{h_{x1}} = M_x \cos(\rho_a) + M_y \sin(\varphi_a) \sin(\rho_a) + M_z \cos(\varphi_a) \sin(\rho_a)$$
$$M_{h_{y2}} = M_y \cos(\varphi_a) - M_z \sin(\varphi_a) \tag{177}$$

The above equation is for 2 axes tilt compensation. However, the plane that is parallel to the Earth switches if the unit is rotated more than 90 degrees on either the pitch or roll axis. Hence, 3 axes tilt compensation is necessary to ensure the proper function of the magnetometer. The rotational matrix of the object can be used to de-rotate the magnetic data. The azimuth can then be used to compare and identified with the result of the yaw rotation calculated from the gyroscope.

There are multiple configurations for implementing an IMU system. The following section gives a brief introduction on the different types of the system as well as the advantages and disadvantages of each configuration.

3.5 Types of Inertial Measurement Units

IMU is an intuitive system, yet very complicated to be implemented. It is a combination of creative hardware design, meticulous calibration techniques, and innovative software designs to reduce the inherit error and achieve higher accuracy of the system. The various configurations listed below are essentially strategies to minimize the shortcomings of the inertial based sensors from the hardware design perspective.

3.5.1 Gimbaled Inertial Platform

The main idea of the gimballed inertial platform is to isolate the inertial sensors inside the inertial frame. The simplest form consists of a triaxial accelerometer and a triaxial gyroscope that are placed within 3 gimbal rings, where each ring revolves around one axis. The outputs of the gyro are connected to a set of servo-motor that drives the gimbals in opposite rotation than the active motion. Hence, it keeps the orientation of the platform constantly aligning with the inertial frame. Because of this unique property, the calculation for translations becomes extremely easy.

On the other hand, the design and the manufacturing of the gears with high tolerance becomes a factor that affects the performance of the stabilizing platform. The unit also requires routine maintenance to assure performance. Gimbal rings based system also suffer from a phenomenon known as gimbal lock, where 2 of the 3 rings are driven in parallel, resulting the loss of 1 degree of freedom. Furthermore, this system contains multiple parts, three gimbals, and 3 servo-motors, which becomes quite bulky for human motion tracking. A gimbaled inertial platform by Marconi electronic systems is shown in Figure 3-7.

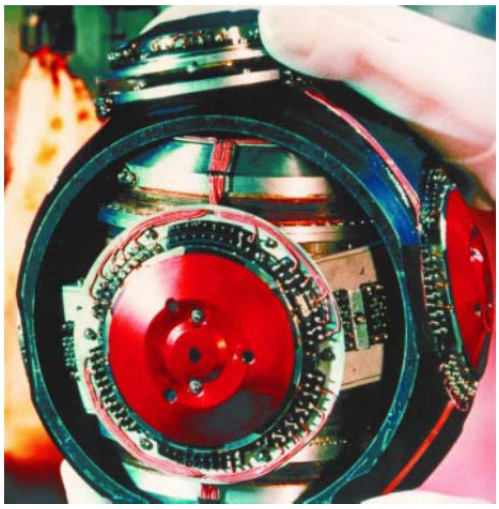

Figure 3-7 - Gimballed inertial platform by Marconi Electronics (now BAE systems) shows the instrumentation are surrounded by 3 gimbal rings

Gimballed inertial platform has recently become popular in aerial photography instead of inertial positioning tracking. A simplified version of the system is used as a means to stabilize the camera, and isolates the maneuvers of the aircraft from the mounting platform.

3.5.2 Strapdown Inertial Measurement Unit System

Strapdown system introduces a new concept of inertial navigation. As mentioned in the gimbaled inertial platform, the system is expensive, bulky and occasionally experiencing gimbal lock issues. A strapdown system fixed both accelerometers and gyroscopes onto the platform and confined both of them in the inertial reference frame. A Strapdown approach eliminates the mechanical rings. It uses the gyroscopes as a way to measure and keep track on orientation of the system instead of canceling the changes mechanically. The gimbals rings functions are performed mathematically from the gyroscopes' outputs. There are two major categories with the

strapdown approach, which the primary difference is the gyroscope that the system uses.

Ring Laser Gyroscope (RLG) strapdown IMU is developed to achieve higher accuracy by eliminating the mechanical aspect of the gyroscope design, where the noise is the primary source of the arithmetic drift. RLG strapdown IMU is still relatively bulky as shown in Figure 3-8; since it needs to house multiple lasers generating units as well as electronics. RLG is typically used in aviation or space flight that requires extreme accuracy.

MEMS based micro-machined gyroscope, on the other hand, is developed to significantly reduce the size of the gyroscope unit. A typical MEMS gyro is 3 by 3 by 1.2 mm and costs less than 5 US dollars. The IMU itself is typically within the size of a 2 inch cube. Figure 3-9 shows an example of the MEMS based IMU. This system is extremely light weighted, low cost, low power and small, thus, the initial framework and testing of the IMU system used in this research is based on the MEMS strapdown IMU system.

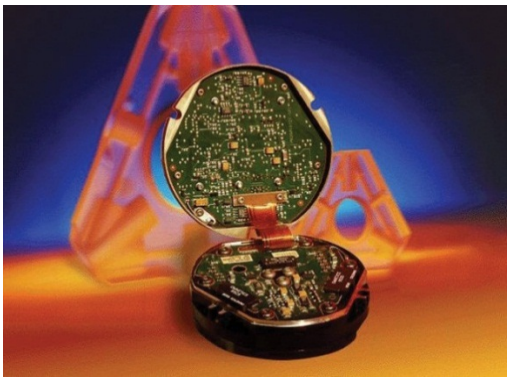

Figure 3-8 - GG1320 ring laser gyros for military aircraft inertial system [74]

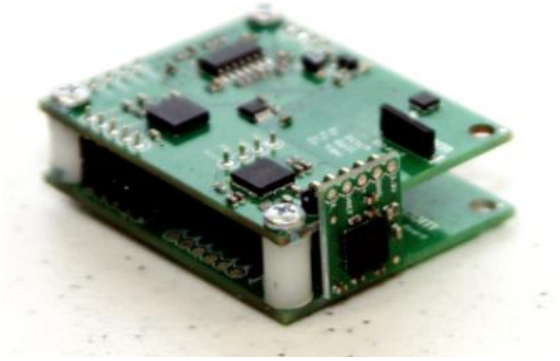

Figure 3-9 - MEMS based strapdown IMU by Sparkfun Electronic

4. Pilot experiment

4.1 Off the Shelf IMU specification for pilot experiment

A pilot study was conducted with off the shell (OTS) hobbyist grade IMU system to check the feasibility of the research, as well as to identify the potential problems. Two IMUs from Sparkfun Electronics were used in the experiments. The basic specification of the unit is shown in Table 4-1.

The unit is processed with LPC2138 ARM7 microcontroller with built in 10 bit analog to digital converter (ADC) for digitization. The data are transmitted via Bluetooth. The data allocation for the transmission protocol is shown in Figure 4-1. As the output from the IMU is in the format of digital code, calibrations for all the sensors are needed to produce meaningful result from the experiment. The calibrations of the sensors are conducted using the protocols outlined in the following sections.

Table 4-1 - Specification of IMU used in pilot study

	Axis	Range	Power	Sensitivity
Accelerometer	X,Y,Z	+/-4g	3.3V	308 mV/g
Gyroscope	X,Y	+/- 500°/s	3 V	2 mV/°/s
Gyroscope	Z	+/- 300°/s	5V	6 mV/°/s
Magnetometer	1	+/- 6 gauss	3 V	1mV/V/gauss
Magnetometer	2,3	+/- 6 gauss	3 V	1mV/V/gauss

Figure 4-1 - Data transmission protocol

4.2 Accelerometer calibration

Accelerometer is calibrated via static calibration. The IMU is hold still by a plastic calibration arm on a flat surface as shown in Figure 4-2 in various positions. The calibration is designed to hold the unit in pitch and roll rotation at 15 degrees of increment. Readouts of the accelerometer in different combination of positions were recorded. The accelerometer raw data are the numerical output from the ADC as shown in Figure 4-3. The figure shows the unit sitting still with Z axis parallel to gravity. Figure 4-4 shows the output data of a full revolution around the x axis. It is obvious that there are offsets and scale factors in the output that requires calibrations.

The accelerometer calibration equation is

$$A_i = \left(a_i + b_{a_i}\right) \times C_a, \qquad C_a = a_range/2^n \tag{178}$$

where i is the x, y, and z axis, A is the corrected acceleration, a is raw acceleration data, n is resolution of the ADC, b_a is the inherit offset, which is the difference between midrange of the ADC $2^n/2$ and the static output of the corresponding axis at 0g, a_range is the full scale range of the of the accelerometer.

Figure 4-5 shows the calibrated static output from Figure 4-3. It shows that Z axis is parallel with the gravity with approximately 9.81m/s output. Figure 4-6 shows the various tilt angles around the x axis after calibration. The tilt angles are calculated based on the equation mentioned from equations (166-168).

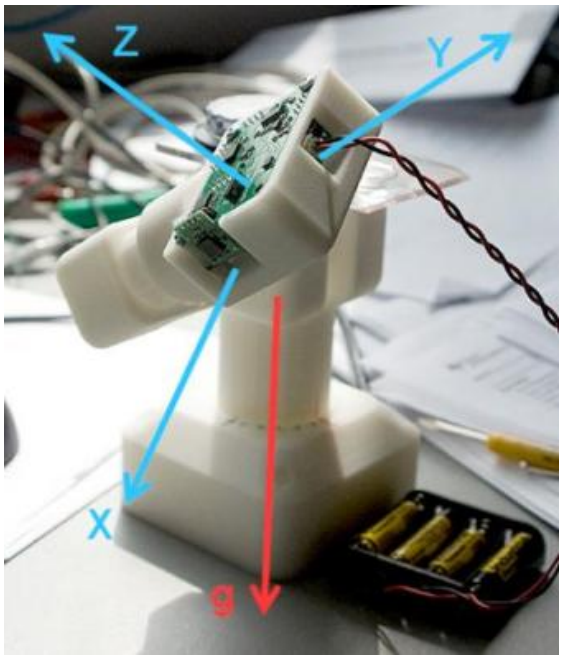

Figure 4-2 - Plastic arm for accelerometer calibration

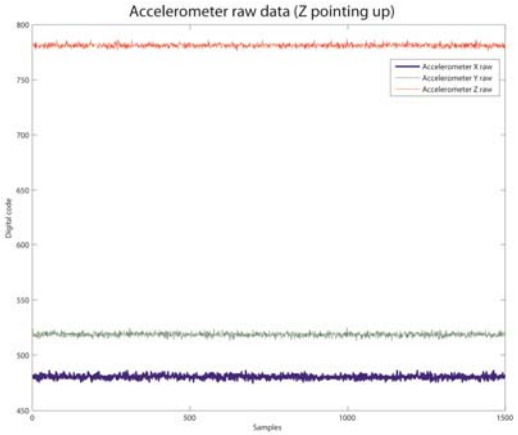

Figure 4-3 - Raw data from the accelerometer

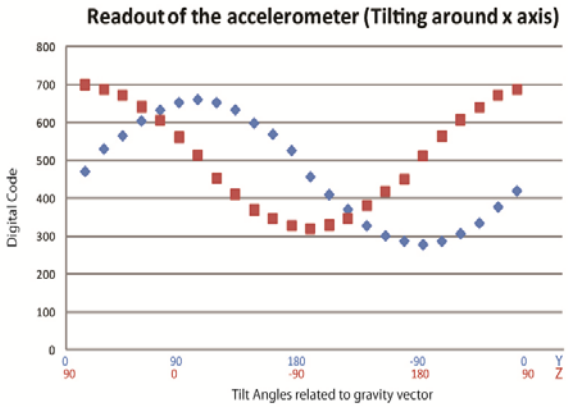

Figure 4-4 - Readout of the accelerometer tilting around the x axis

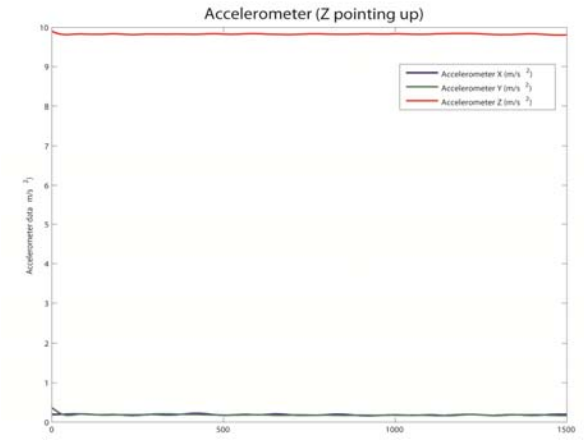

Figure 4-5 - Calibrated static output

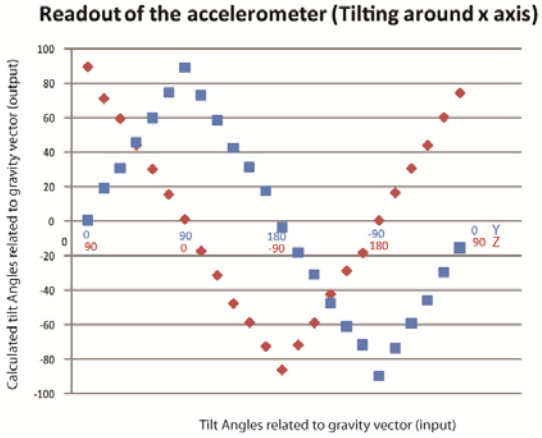

Figure 4-6 - Calibrated tilt angle output of Y and Z around the x axis

4.3 Gyroscopes Calibration

Gyroscope, unlike accelerometer, measures the changes within the inertial frame. Hence, dynamic calibration is required. Due to limit access to rate table equipment, the gyroscopes are calibrated with a servo turn table. Since the turn table can only produce rotation in 1 axis, the calibration arm is use to align different axes of the IMU to the turn table's rotational axis. The gyroscopes are first hold still on a flat service to determine its bias. The raw data from the gyroscopes of a stationary IMU is shown in Figure 4-7.

The gyroscope output is very stable, however it has a considerably amount of noise. The turn table is able to provide up to 136°/s rotation at 8°/s increment on both clockwise and counter clockwise directions. The output of the gyroscope rotating around the x-axis with different turning rate is shown in Figure 4-8.

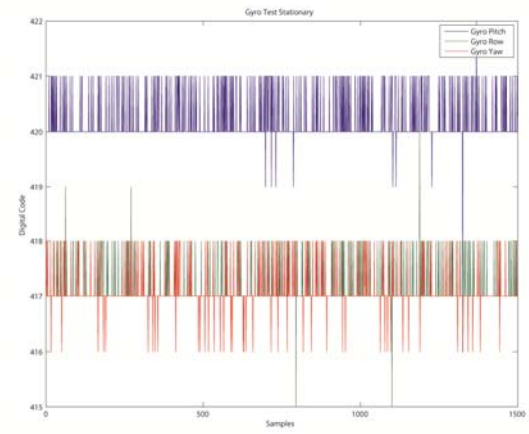

Figure 4-7 - raw data from the gyroscopes

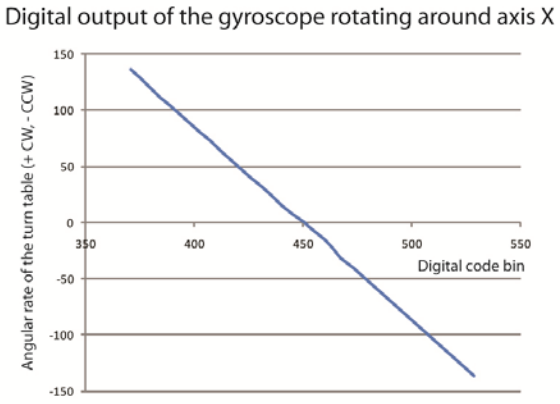

Figure 4-8 - Raw data of the gyroscope at different turning rate of the turn table around x axis

The calibration equation for gyroscope is similar to the accelerometer, which is

$$W_i = \left(w_i + b_{g_i}\right) \times G_{scale}^{i} \tag{179}$$

where i is the x, y, and z axis, W is the corrected angular velocity, w is raw gyroscope data, n is resolution of the ADC, b_g is the inherit offset, which is the difference between midrange of the ADC $2^n/2$ and the static output of the corresponding axis when it is stationary, G_{scale}^{i} is the calibration scale of the gyroscope which is defined by the following equation,

$$G_{scale}^{i} = \frac{\overline{w_j / \left(\frac{int(\overline{w_{out_j}^{i}} - \overline{b_{g_i}})}{int(\overline{w_{out_j}^{i}} - \overline{w_{out_{j-1}}^{i}})} \right)}}{\overline{int(\overline{w_{out_j}^{i}} - \overline{w_{out_{j-1}}^{i}})}} \tag{180}$$

where i = x, y, and z axis, G_{scale} is the scale for the gyroscope in axis l, w_i is the rate of the turn table at angular velocity j, and $w_{out_j}^{l}$ is the digital output of the gyroscope on l axis at angular velocity j.

4.4 Magnetometer Calibration

Magnetometer is calibrated by simply revolving the IMU system around each of its axis. Data are collected by rotating the magnetometers around the plane perpendicular to gravity around each axis. The scale factors for the axis perpendicular to the rotation can be determined by,

$$M_{SF(1)} = \frac{(\max(m_2) - \min(m_2))}{(\max(m_1) - \min(m_1))} \tag{181}$$

$$M_{SF(2)} = \frac{(\max(m_1) - \min(m_1))}{(\max(m_2) - \min(m_2))} \tag{182}$$

The offsets can be calculated by

$$M_{Bias(i)} = \left(\frac{(\max(m_i) - \min(m_i))}{2} - \max(m_i)\right) M_{SF(i)}, \qquad i = 1{:}3 \tag{183}$$

Figure 4-9 shows the deviation angle between the IMU and the magnetic north. The non-linear portion is attributed to the ferromagnetic hysteresis.

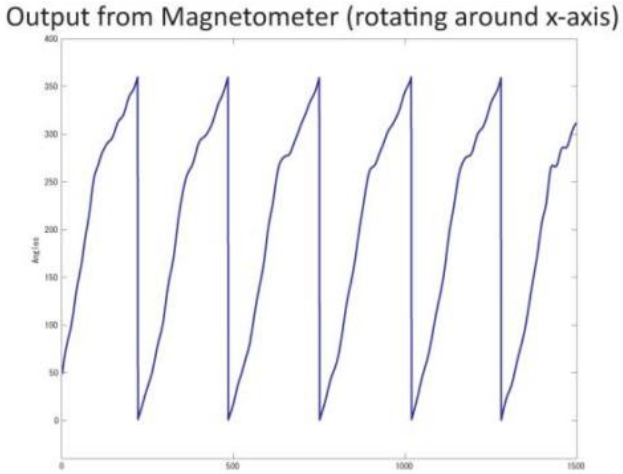

Figure 4-9 - Deviation angle between the sensors to true north calculated from magnetometer rotating around x-axis

4.5 Attitude Estimation Experiment

Quaternion was chosen to be used as the orientation representation because it has fewer elements and constraints, yet it contains no singularity point. Discrete Kalman filter and EKF are implemented for the determination of orientation using these sensors. The *priori* estimates are calculated by the inputs from the gyroscopes, while the *posteriori* estimates are determined by the calculation from the accelerometers for pitch and roll and magnetometer for yaw. The implementation of these algorithms is discussed in detail in chapter 6. The basic process flow is shown in Figure 4-10.

The orientation outputs are projected back as Euler angles for visualization. Figure 4-11 and Figure 4-12 show the orientation outputs of the IMU undergoing the same activity of rotating around the z-axis for Kalman and Extended Kalman filters respectively.

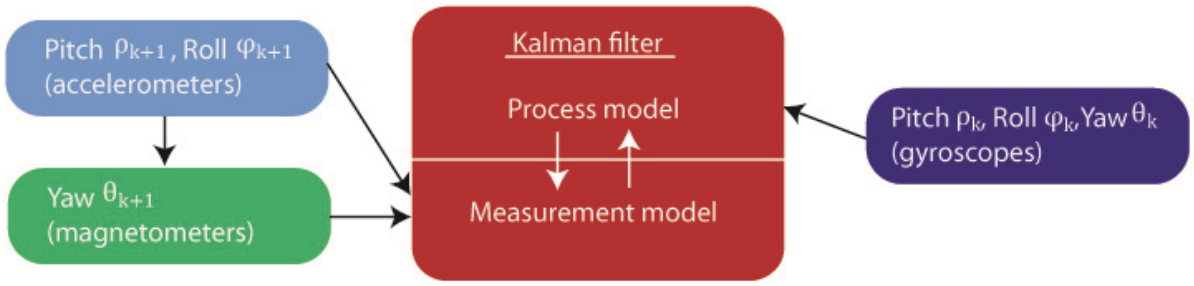

Figure 4-10 - Kalman filter models for IMU

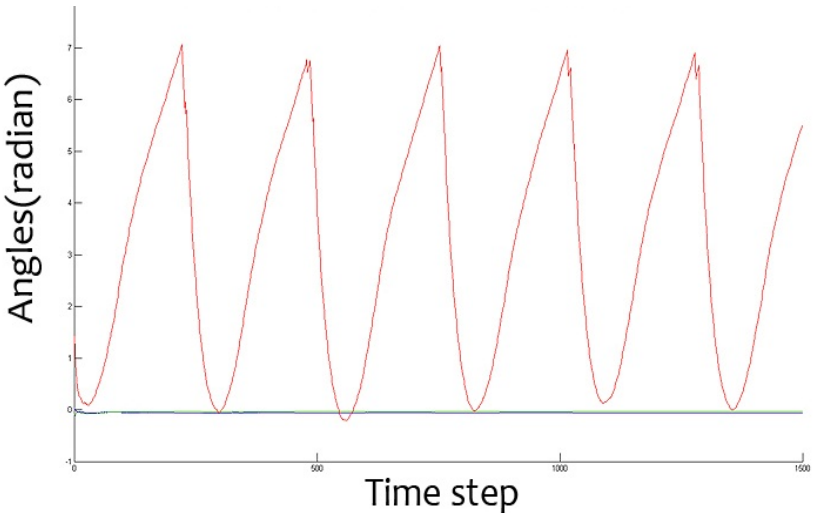

Figure 4-11 - Output after Kalman filter

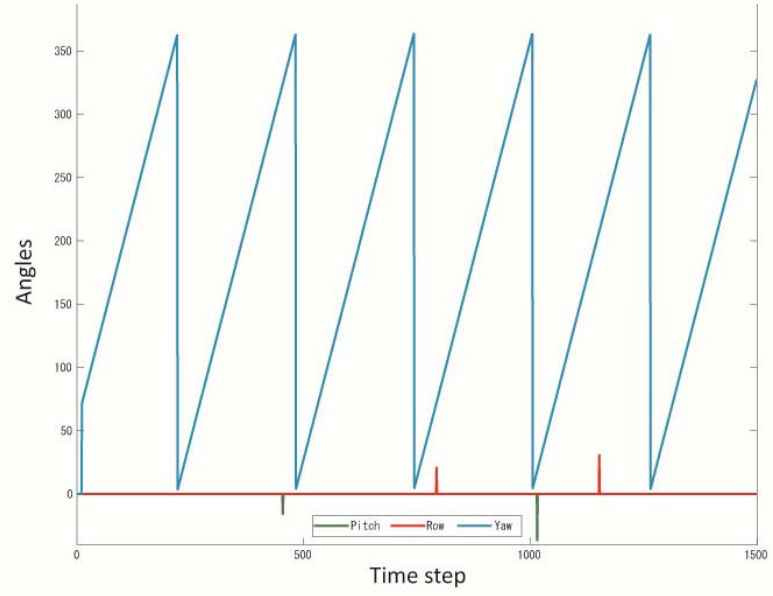

Figure 4-12 - Output after EKF

4.6 Human body motion tracking

The next step of the pilot experiment is to test the feasibility for human motion tracking. An IMU containing knee brace was made with a prototyping machine and two IMUs are secured onto the brace as shown in Figure 4-13. Two passive optical tracker targets are attached to the brace for comparisons. The infrared optical camera is placed such that the knee joint is within the viewing volume of the unit.

The first sets of test are static testing, which include sitting and standing still. The data for standing and sitting postures, as shown Figure 4-14 a and b, were recorded for 1 minute for each posture. Due to the differences in sampling rate between the IMU and the optical unit, the IMU orientations after calculation are re-sampled to fit with the optical data time frame. The data for standing still is shown in Figure 4-15 for thigh and Figure 4-16 for shank. The output orientations for sitting are shown in Figure 4-17 for thigh and Figure 4-18 for shank. There is no post processing on the optical data except converting the absolute orientation into relative orientation to its initial position at (0^o, 0^o, 0^o). The root mean squared errors (RMSE) between the IMU output and the optical tracker for static testing are calculated and summarized in Table 4-2.

Figure 4-13 - IMU brace

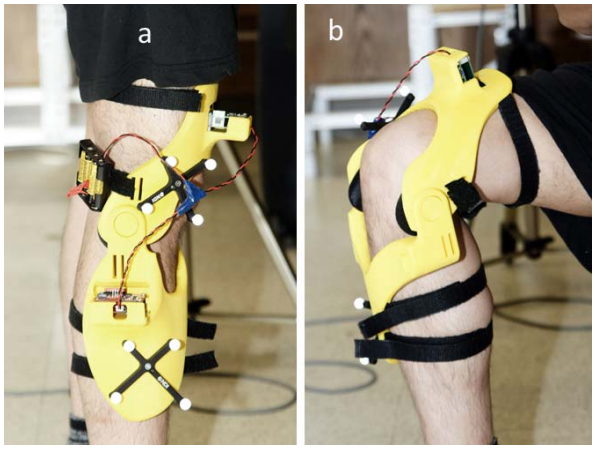

Figure 4-14 - Static testing with IMU brace. (a) Standing still; (b) Sitting still

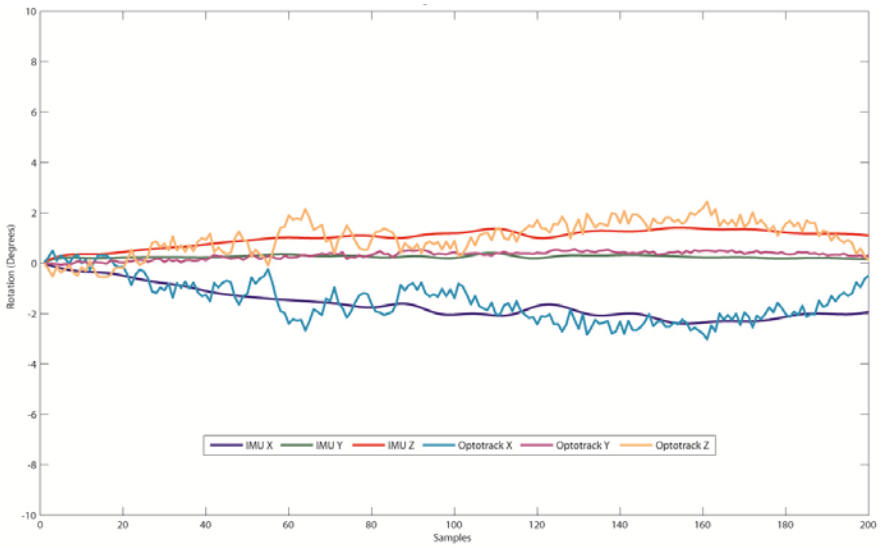

Figure 4-15 - IMU vs optical outputs (Standing, Thigh) Light blue: Optical X, Purple: Optical Y, Yellow: Optical Z; Dark blue: IMU X, Green: IMU Y, Red: IMU Z

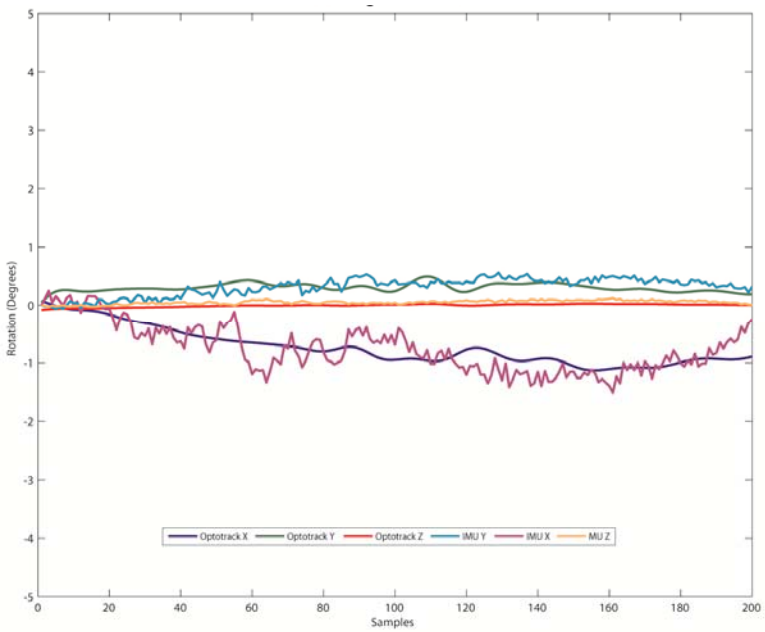

Figure 4-16 - IMU vs optical outputs (Standing, Shank) Dark blue: Optical X, Green: Optical Y, Red: Optical Z; Purple: IMU X, Light blue: IMU Y, Yellow: IMU Z

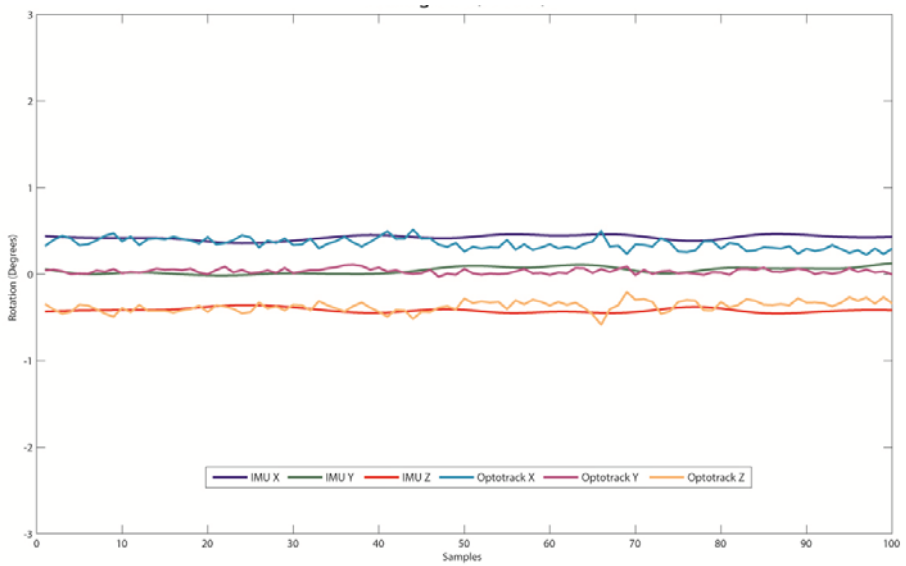

Figure 4-17 - IMU vs optical outputs (Sitting, Thigh) Light blue: Optical X, Purple: Optical Y, Yellow: Optical Z; Dark blue: IMU X, Green: IMU Y, Red: IMU Z

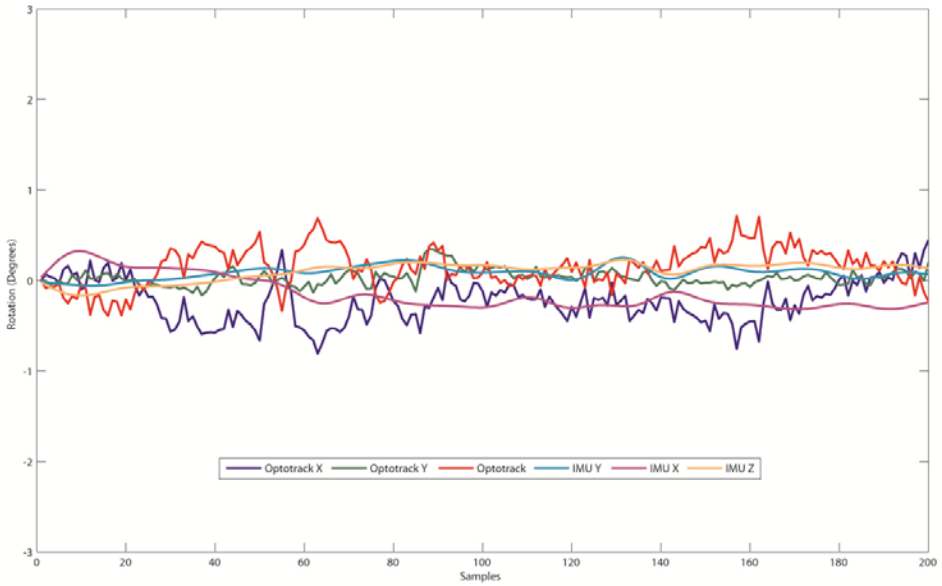

Figure 4-18- IMU vs optical outputs (Sitting, Shank) Dark blue: Optical X, Green: Optical Y, Red: Optical Z; Purple: IMU X, Light blue: IMU Y, Yellow: IMU Z

The dynamic testing of the brace is conducted by the following two activities, deep knee bend and chair rise as shown in the sequence in Figure 4-19 and Figure 4-20 respectively. Multiple of these activities were performed under the surveillance of both IMU and optical tracking systems. The orientation of the IMU and optical unit were both calculated relative to their respective initial null orientation. Figure 4-21 and Figure 4-22 show the comparisons between the IMU calculation and the optical tracker of the thigh and shank respectively for deep knee bend. Figure 4-23 and Figure 4-24 show the data during chair rise activities for the thigh and shank respectively. The RMSE between the IMU output and the optical tracker for dynamic testing are calculated and summarized in Table 4-2.

Figure 4-19 - Deep knee bend sequence

Figure 4-20 - Chair rise sequence

Table 4-2 – RMSE on each axis during static and dynamic testing

	X Axis	Y Axis	Z Axis
Thigh (Static)	0.91°	0.29°	1.30°
Shank(Static)	0.30°	0.15°	0.36°
Thigh (Dynamic)	3.86°	3.84°	4.09°
Shank (Dynamic)	3.41°	4.51°	3.06°

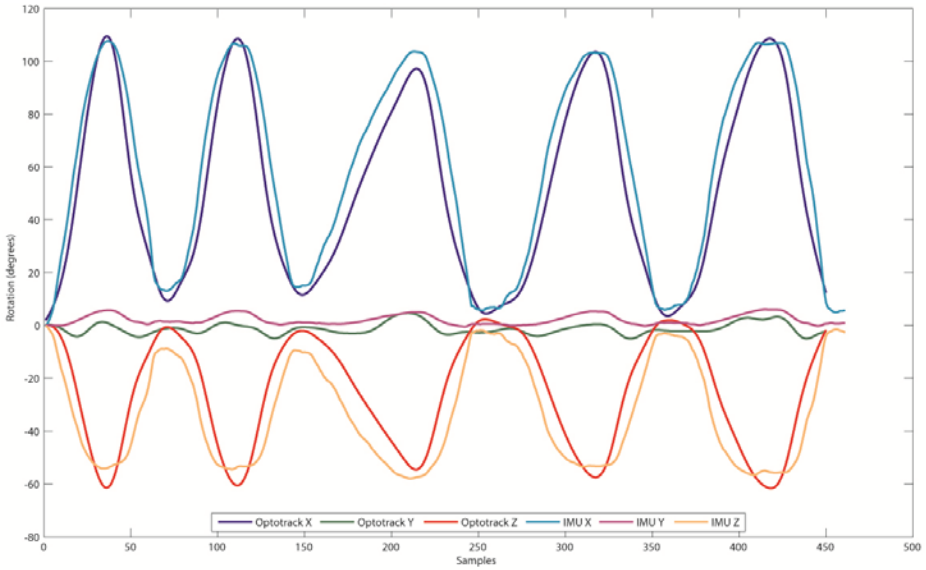

Figure 4-21 - IMU vs optical outputs (Deep Knee Bend, Thigh). Dark blue: Optical X, Green: Optical Y, Red: Optical Z; Light blue: IMU X, Purple: IMU Y, Yellow: IMU Z

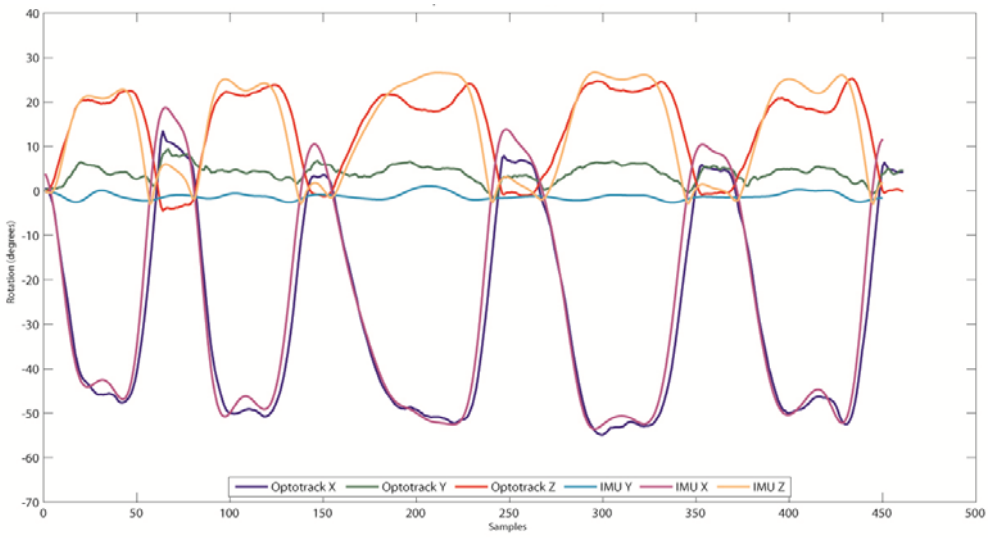

Figure 4-22 - IMU vs optical outputs (Deep Knee Bend, Shank). Dark blue: Optical X, Green: Optical Y, Red: Optical Z; Purple: IMU X, Light blue: IMU Y, Yellow: IMU Z

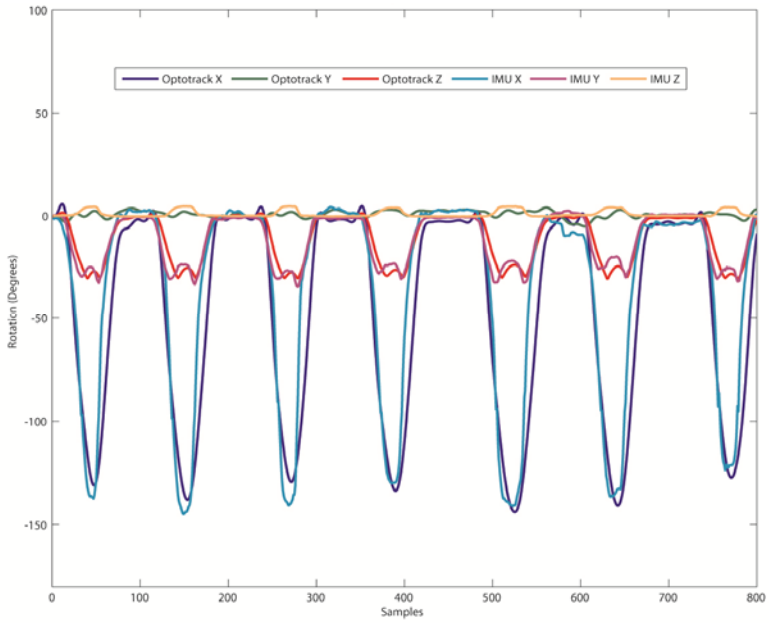

Figure 4-23 - IMU vs optical outputs (Chair RiseThigh). Dark blue: Optical X, Green: Optical Y, Red: Optical Z; Light blue: IMU X, Purple: IMU Y, Yellow: IMU Z

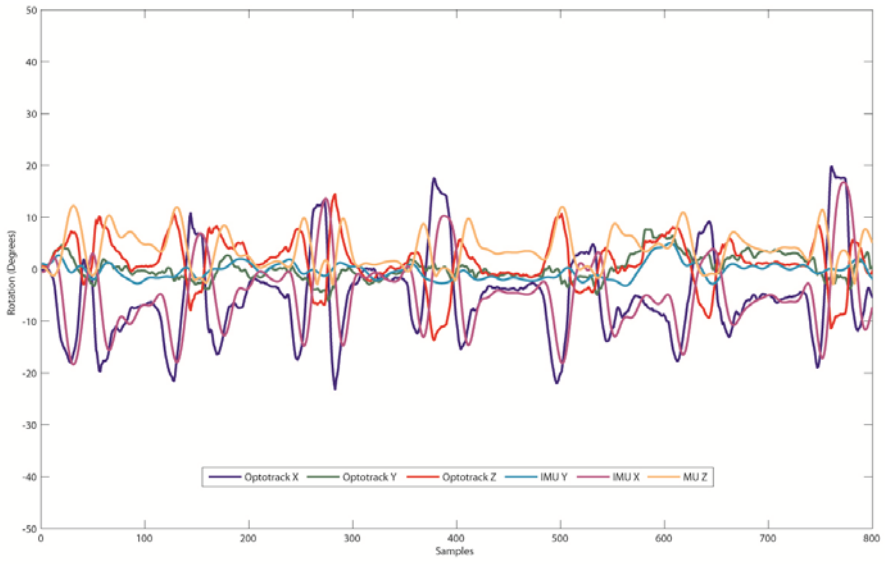

Figure 4-24 - IMU vs optical outputs (Chair Rise, Shank). Dark blue: Optical X, Green: Optical Y, Red: Optical Z; Purple: IMU X, Light blue: IMU Y, Yellow: IMU Z

The RMS error shows a significant difference in estimation calculations when compare between static and random dynamic activities. There are several factors that contribute to the error. The most dominate source of error comes from the limited resolution of the ADC. At 10 bit, the theoretical maximum resolving power of the least significant bit (LSB) on the ADC on each sensor is shown in Table 4-3. This, however, does not include practical practice such as limiting the sensor output range to be smaller than the maximum dynamic range of the ADC. The inherit limit of the ADC introduce a considerable amount during error to the signal.

Another source of error originated from the gyroscopes. Duration calibration, the outputs of the gyroscopes were observed at different angular velocities of the rotating table. Figure 4-25 shows the box plot of the outputs from one of the axis. As the rotating speed increases, the variance of the gyroscopes from the OTS IMU increases substantially. This leads to an unexpected rate dependent error in the error covariance of the system. In addition, variance of the signal behaves as a non-linear function.

The third source of error comes from the Kalman filter algorithm used for sensor fusion. Because quaternion is a hyper-complex vector with a unit-norm constraint resides on a different manifold, the error covariance matrixes used in the algorithm is an approximation which appears to underestimate the true error covariance.

Table 4-3 - Resolving power of the lease significant bit on each sensor

	LSB
Accelerometer	7.66cm/s
Gyroscope (X,Y)	0.98^{o}/s
Gyroscope (Z)	0.59^{o}/s
Magnetometer	11.72mgauss/s

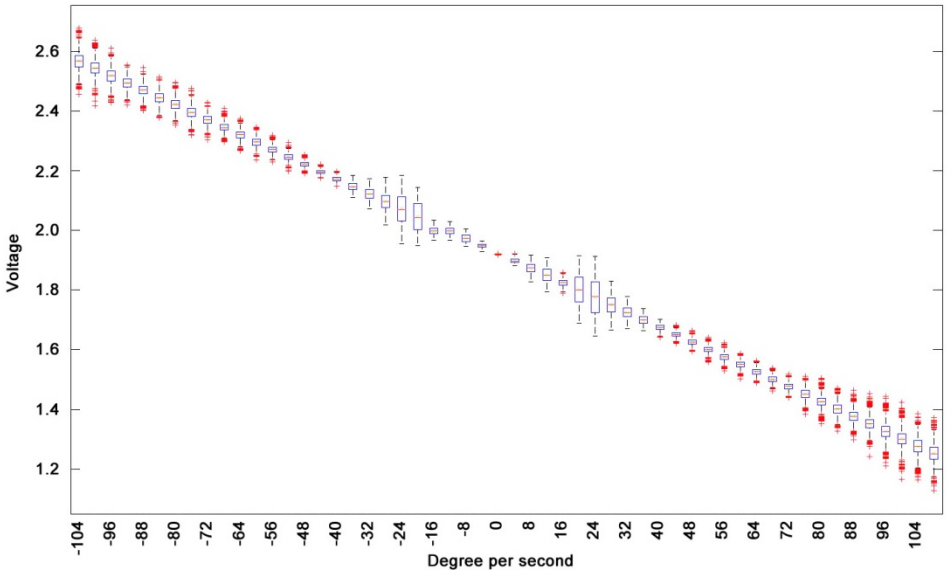

Figure 4-25 - Box plot of the X-axis of the gyroscopes undergoing rotations at different angular velocities

5. Hardware implementation

There are three major areas of improvement based on the analysis from the OTS IMU system, which are the ADC resolution, the sensors selection and the attitude estimation algorithm design. This chapter focuses on the hardware implementation and introduces a modular design to enhance flexibility to the IMU.

5.1 Hardware specification

After examining the performances of the gyroscopes from several manufacturers, LPY510-AL and LPR-510-AL by ST Microelectronics were chosen due to their superior performance as shown in Figure 5-1. Both of the sensors have two operating configurations that can be adjusted by an external signal. The sensors selected in the prototype design are shown in Table 5-1. Low power microcontroller (MSP430F2274, Texas Instrument) was used as central processor. A compact wireless transmitting module (A2500R24A, Anaren) was used for communication. The receiver uses pre-determined device identification number to process the received data from multiple IMUs. The high level system architecture is shown in Figure 5-2.

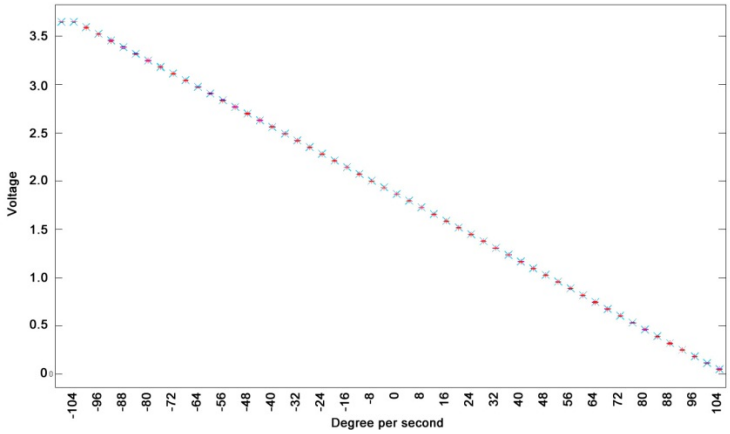

Figure 5-1 Box plot of the X-axis of the gyroscopes (LPR510-AL) undergoing rotations at different angular velocities

Table 5-1 - Inertial sensors used in prototype

	Model	Axis	Range	Power	Sensitivity
Accelerometer	MMA7361L	X,Y,Z	+/- 1.5g	3.3V	800 mV/g
Gyroscope (1)	LPR510-AL	X,Y	+/- 100°/s	3V	10 mV/°/s
Gyroscope (1)	LPY510-AL	Z	+/- 100°/s	3V	10 mV/°/s
Gyroscope (2)	LPR510-AL	X,Y	+/- 400°/s	3V	2.5 mV/°/s
Gyroscope (2)	LPY510-AL	Z	+/- 400°/s	3V	2.5 mV/°/s
Magnetometer	HMC1053	X,Y,Z	+/- 6 gauss	3V	1mV/V/gauss

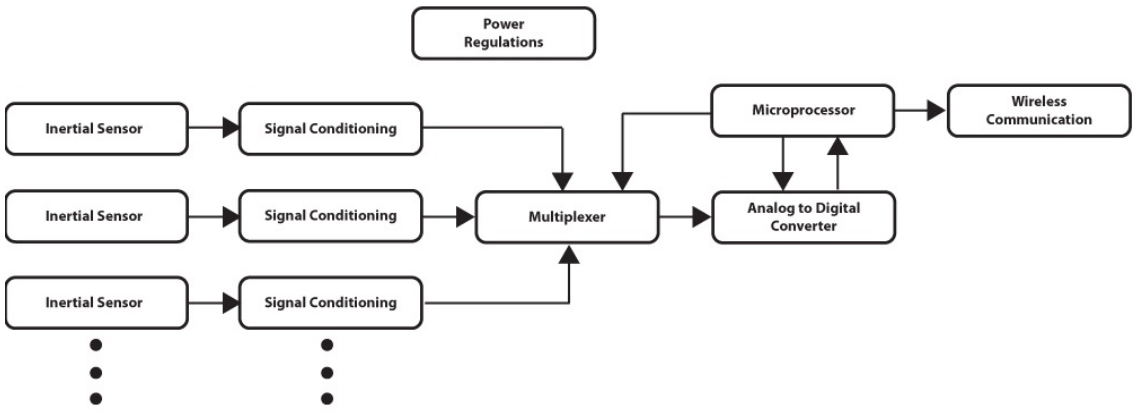

Figure 5-2 - System Architecture for Prototype IMU

In the OTS IMU, the signal to noise ratio of a 10 bit converter operating at its fastest convention speed has approximately 55dB dynamic range, which is inadequate for a system demanding high accuracy. A 16-bit A/D converter, which gives approximately 98dB signal dynamic range, should be sufficient for the system. However, in practice, the actual effective number of bits (ENOB) is much lower than 16-bit at the fastest conversion speed of the ADC due to various design limitations such as the settling time of the converter. In addition, the rail-to-rail input signal clearance is required to avoid undesired signal chopping. Because of these considerations, a multi-channel 24-bit A/D converter (ADS1258, Texas Instrument) was selected. At its fastest convention speed for single ended operation and optimal sensor range, the converter has an approximately 18 noise free effective bit.

5.2 Power Supply Design

The IMU requires two power source regulators for components operating at 3.3V and 5V. A high capacity capacitor is used as the decoupling capacitor for the 5V power regulator to provide more stable voltage reference for the high resolution ADC. The schematic design is shown in Appendix A. The circuit is designed to be powered by either 2 CR2052 coin cell batteries or a 200mAh Lithium Iron Phosphate rechargeable battery, which can last approximately two hours.

5.3 Signal conditioning circuit Design

The primary goal of the signal conditioning circuit is to minimize the difference between the sensors output range and the ADC input range. Figure 5-3 is an offset amplifier designed for the signal outputs from the accelerometers and gyroscopes. The value of the resistors network can be determined by the following equations.

$$m = \frac{Vdd_{ADC} - Vss_{ADC}}{V_{hi_{sensor}} - V_{low_{sensor}}} \tag{184}$$

$$b = (Vdd_{ADC} - Vss_{ADC}) - V_{hi_{sensor}} \times m \tag{185}$$

where Vdd_{ADC} is the maximum allowable input to the ADC, Vss_{ADC} is the minimum allowable input to the ADC, $V_{hi_{sensor}}$is the maximum output range from the sensor and $V_{low_{sensor}}$ is the minimum output range of the sensor. RF and R2 can then be calculated after picking the resistor value for R1 and RG.

$$R_F = (m - 1) \times R_G \tag{186}$$

$$R_2 = \frac{R_1}{\left(1 - \frac{|b|}{Vcc} \times \frac{R_G}{R_F}\right) \Big/ \left(\frac{|b|}{Vcc} \times \frac{R_G}{R_F}\right)} \tag{187}$$

The resistors value for the signal conditioning circuit for each type of sensors are determined from circuit simulation as shown in Figure 5-4.

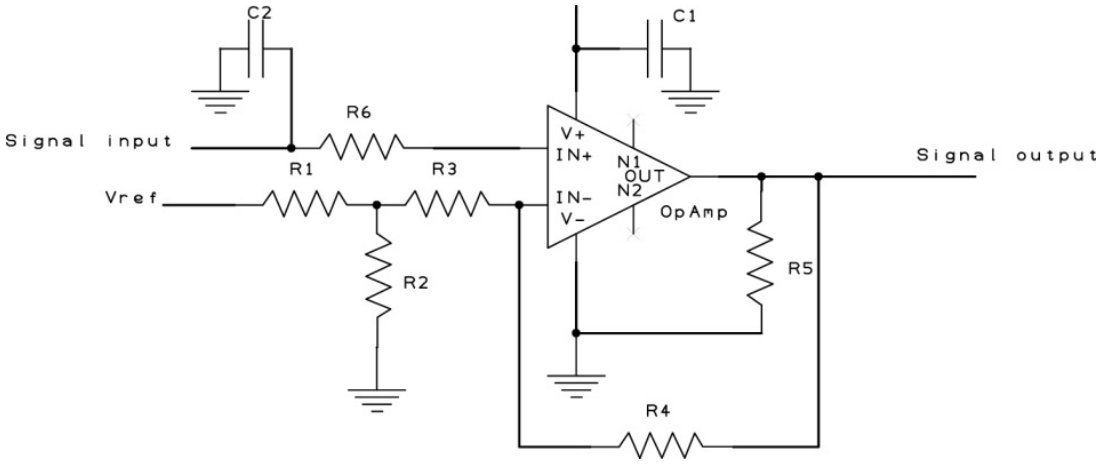

Figure 5-3 - Signal conditioning circuit for accelerometers and gyroscopes

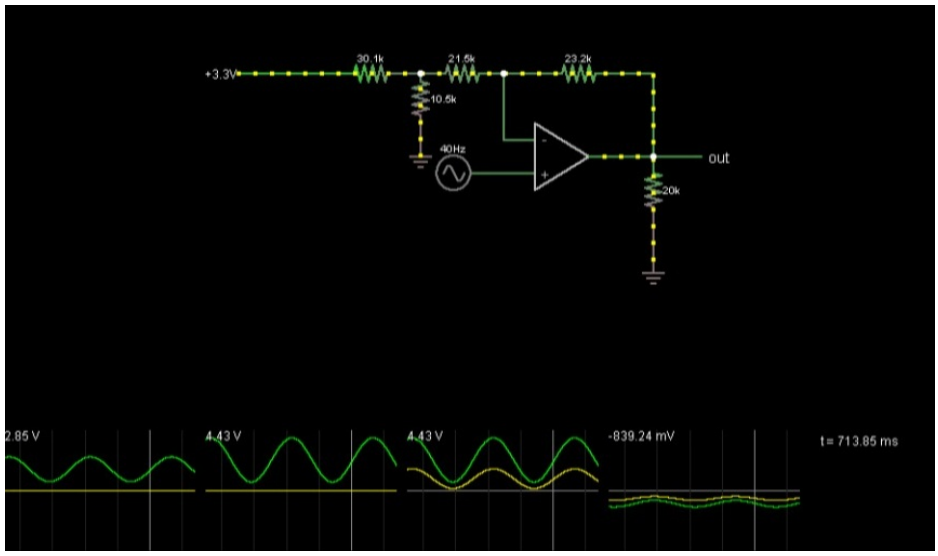

Figure 5-4 - Circuit simulation for the signal conditioning circuit

The magnetometers used in the prototype are magnetoresistive sensors (HMC1053, Honeywell, Inc). The sensors use an internal Wheatstone bridge for the magnetic field measurements. The outputs of the bridge are input into a differential amplifier as shown in Figure 5-5. Because the output of the differential amplifier is bi-polar, a stable offset voltage is applied to the input at the positive end of the sensor output.

A set and reset circuit is used to eliminate any prior magnetic disturbance. This is achieved by sending a positive and negative current pulse to the sensors. The pulse sequence is controlled by the microcontroller. The schematic designs of the signal conditioning circuits are shown in Appendix A.

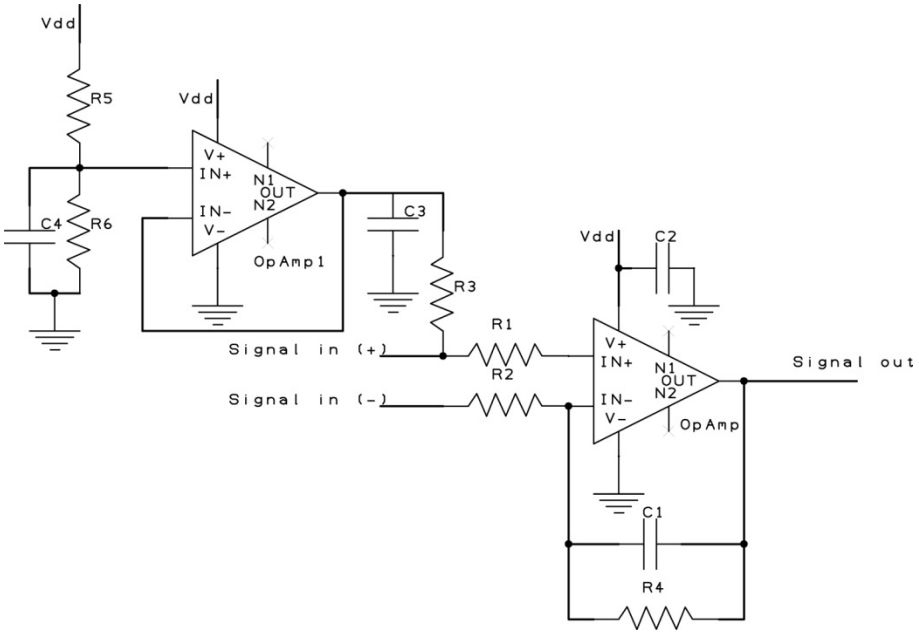

Figure 5-5 – Signal conditioning circuit for Magnetometers

5.4 Electronics Design

The outputs from the signal conditioning circuit are connected directly into ADS1258, where they are multiplexed internally to the ADC. The output of the ADC is a 32 bit word, which includes an 8 bit header to indicate the active channel during conversion. The data is collected with the microcontroller (MCU) MSP430 via spi-by-wire, breaks up into packets, and submits to the transmitter (TX) via universal asynchronous receiver/transmitter (UART). The pin routing table for the MCU is shown in Table 5-2. The schematic designs of the signal conditioning circuits are shown in Appendix A.

5.4.1 Firmware Design

The firmware for the IMU is designed to co-ordinates with all other electronics of the system, and is stored on the MCU. The firmware first initializes operating instructions to the ADC and the TX. The unit then enters seeking mode to look for the access point (AP). To conserve power, the data acquisition only begins after the IMU has joined a network. TX is disabled during the ADC convention period and re-enable when data is ready to transmit. The IMU rejoins the shared network and transmits the data packet. The instructions flow of the MCU is shown in Figure 5-6 and Figure 5-7.

The access point composes of a MCU and a receiver (RX). The primarily function of the access point is to monitor and establish connection to the end devices within its broadcasting vicinity. It also performs checks on the received data to ensure their integrity and transmits them to the computer. The firmware instruction flow of the access point is shown in Figure 5-8.

The dataset from each acquisition is broken into two transmission packages. The received package on the access point formats the data for PC communication as shown in Figure 5-9. The data in each package is encapsulated within the Beginning of Data

(BOD) and the End of Data (EOD) tags. The access point also assigns an end device number according to the sequence that the units join the network. An additional data sequence number is used to distinguish the two data packages. The 8-bit channel identification number in the original data is discarded. The channel data are converted into a 6-digit hex number. The device identification number was programmed uniquely into each IMU such that the processing algorithm can locate the stored calibration data.

Table 5-2 - Routing table from Microcontroller

Connection to Dongle	MISO	P 3.5
	MOSI	P 3.4
Connection to ADC	DRDY	P 1.0
	MISO	P 3.5
	MOSI	P 3.4
	CS	P 1.4
	SCLK	P 3.0
	PWDN	P 1.1
	RESET	P 1.2
	START	P 1.3
Connection to CC2500	GD0	P 2.6
	GD2	P 2.7
	MISO	P 3.2
	MOSI	P 3.1
	CS	P 4.5
	SCLK	P 3.3

Note: MISO=Master In Slave Out; MOSI=Master Out Slave In; DRDY=Data ReaDY; CS=Chip Select; SCLK=Serial CLocK; GD0 and GD2= interrupt flag

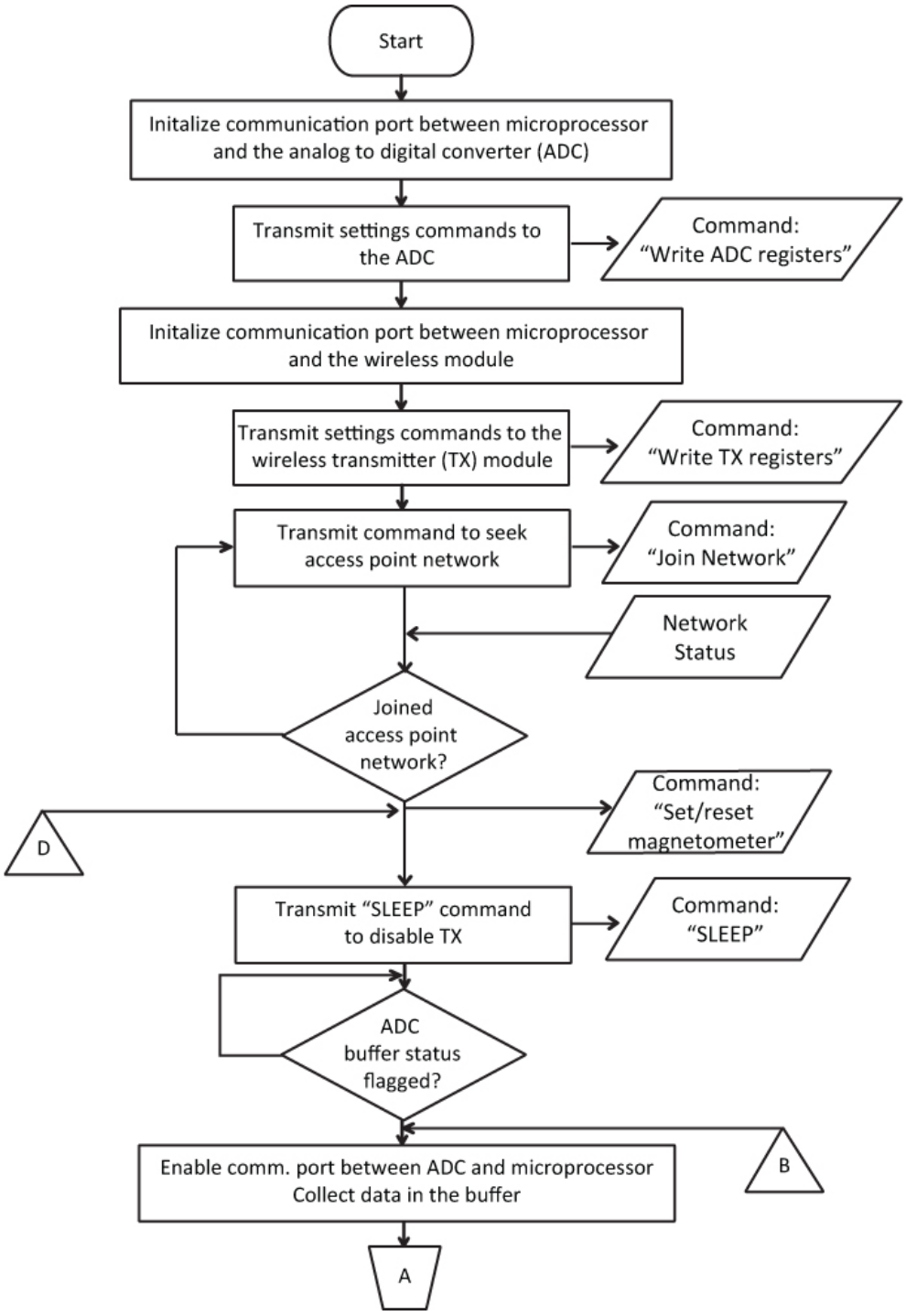

Figure 5-6 - Firmware instruction flow for end devices (A)

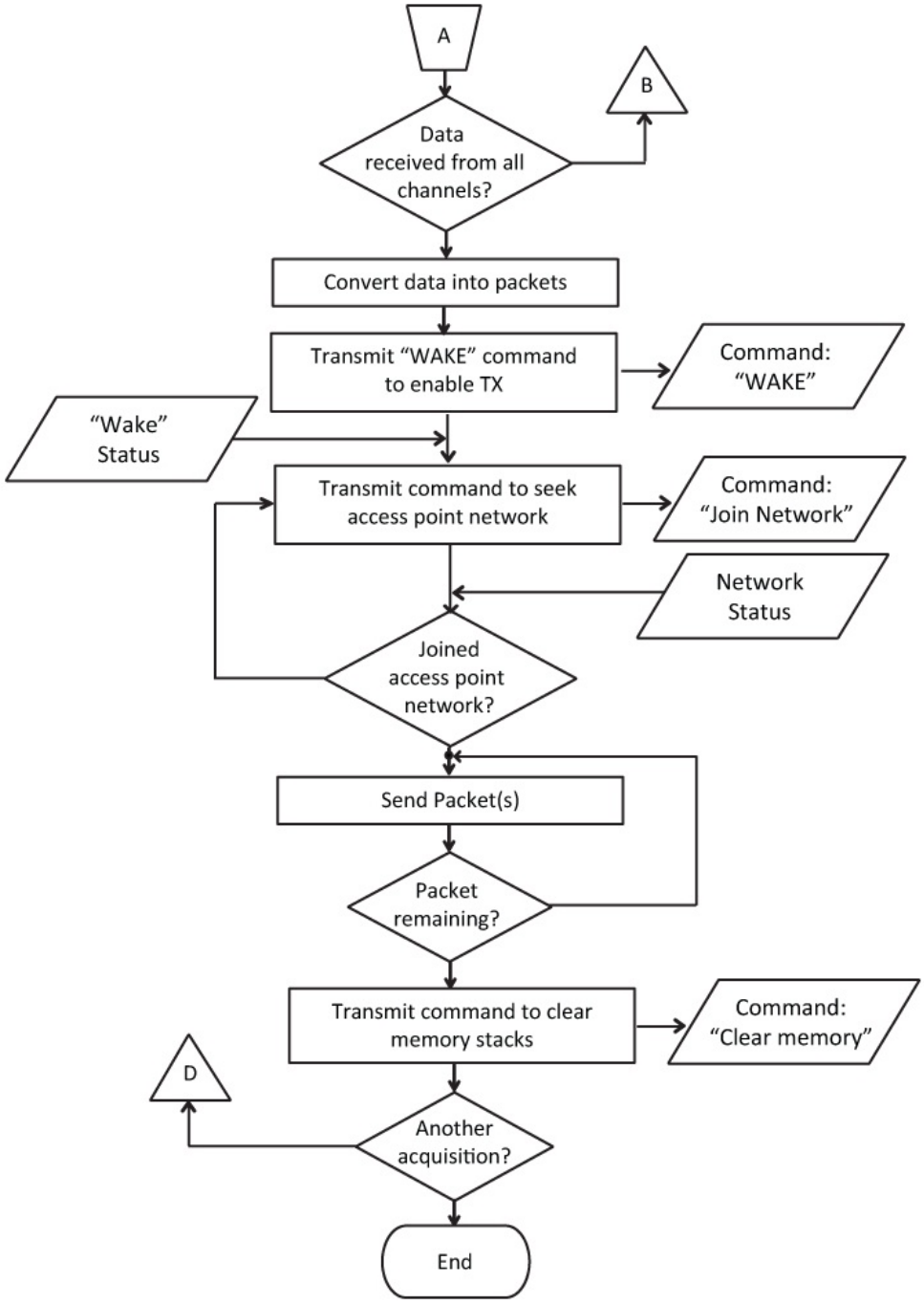

Figure 5-7 - Firmware instruction flow for end devices (B)

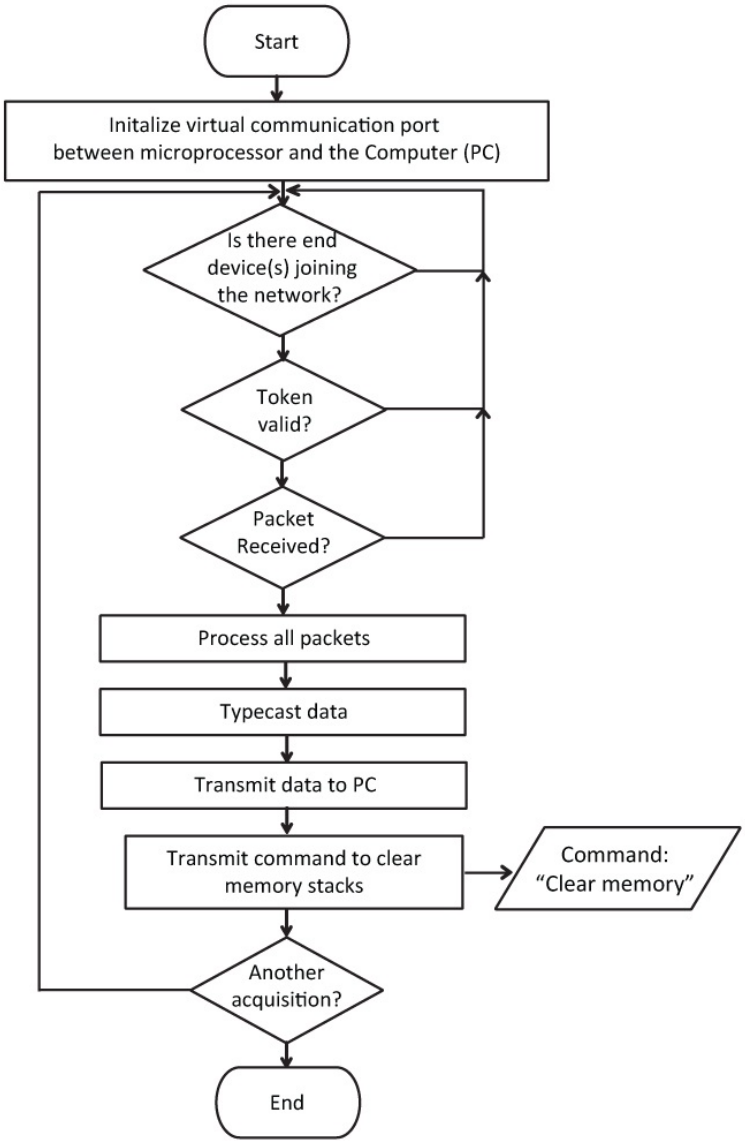

Figure 5-8 -Firmware instruction flow for Access Point

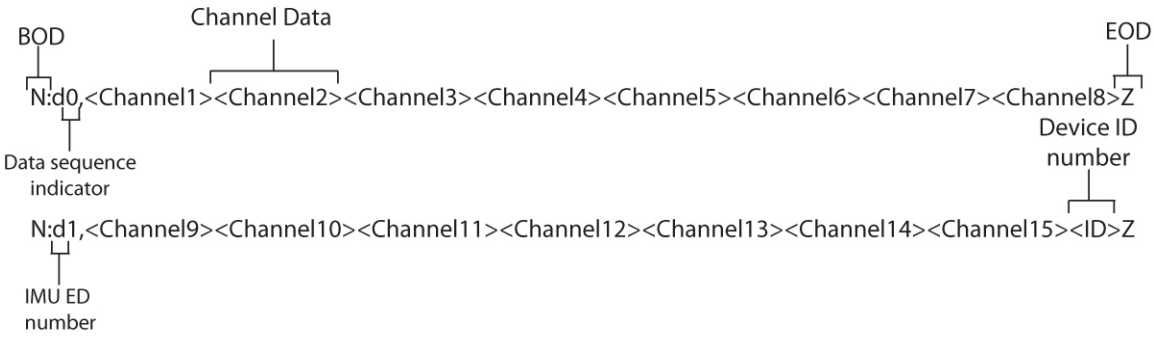

Figure 5-9 – Wireless transmission data format

5.5 Modular IMU design

One of the observations from working with the OTS IMU is that the system architecture of these units is rigid and very limiting. The specifications of the unit are fixed by the sensors that cannot be easily modified and substituted. While it is obvious that there is a difference on the range of motion among different joints of the human body, the degree and magnitude of the motion also varies with activities. In a control environment, the motions are predicable, and corresponding IMUs can be used to access motion. However, human activities are very different than missiles, airborne vehicles and automobiles. It contains agile transition motions, such as switching from walking to running, or sudden change of direction similar to taking sharp corner turns while in motion. A fixed set of IMU sensors creates excessive boundaries that hinder the performance of the system.

With the consideration of designing a tracking device that needs to be used and operates in different conditions, a modular strategy is implemented. The principle of this design is to allow sensors with different specifications to connect to the same system in a 'plug-and-play' manner. The IMU system is separated into electronics and sensing

components, which are attached together via inter-board connectors. This allows the users to switch sensors with different sensitivities based on their needs and applications.

5.5.1 Circuit layout

The circuit layout of the IMU is designed with Easy-PC by Number One system. Several design iterations were made to test different components and features, as well as improving the system with different mix-signal layout schemes (Figure 5-10). The latest version (2.4.2) uses a star-ground configuration on a 4 layers design. It contains three sensor ports with two of the ports support 6 input channels, and last one supports 3 channels. The current configuration groups the accelerometers and gyroscopes into one sensing strip, and the magnetometer is placed on a separate one to minimize interference (Figure 5-11 A). In addition, two configurations of the gyroscopes with different sensitivities were implemented. The assembled unit is shown in (Figure 5-11 B). The current design is 35.5x35.6x13.7 mm in size. The layout of the current circuit design can be found on Appendix B.

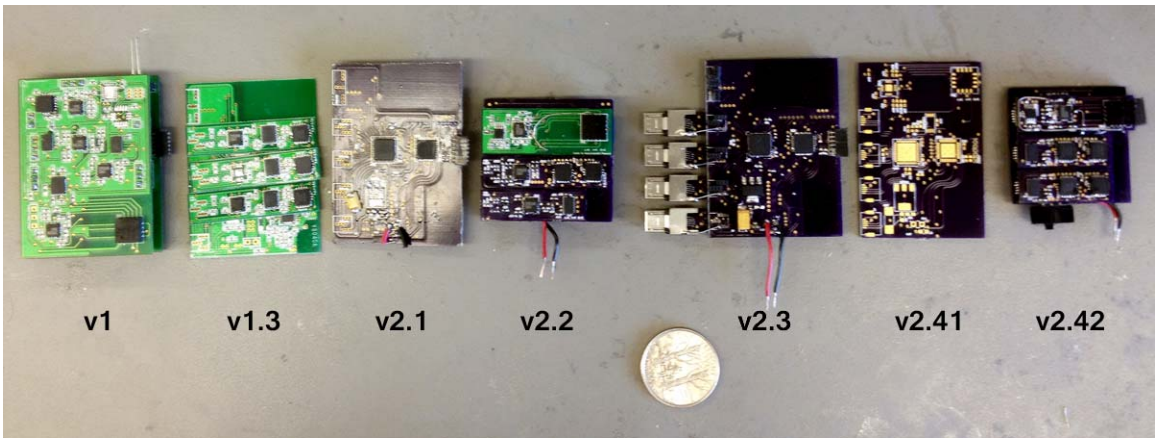

Figure 5-10 - The evolution of the IMU circuit designs

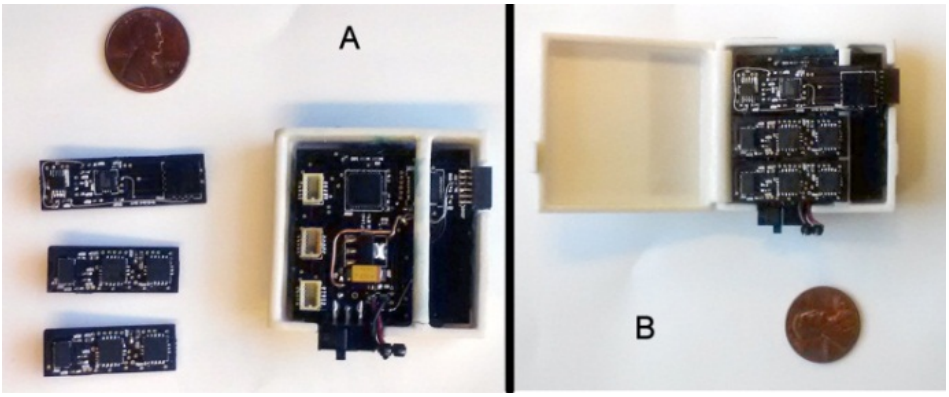

Figure 5-11 – The sensing strips and electronic board of the modular IMU design (A). The assembled IMU system (B)

6. Software Implementation

In chapter 2, several basic principles of the attitude estimation algorithms were discussed. This chapter is going to focus on the implementation of these algorithms for IMU tracking application. As mentioned previously in Chapter 2 and 4, quaternion is an orientation representation of the SO(3) rotation group that does not contain singularity point. Hence, the following algorithms are implemented with quaternion.

6.1 Kalman Filter Implementation

Kalman filter (KF) is a generic model for estimation applications. One of the most notably application is the tracking problem. Early implementation of KF was used for military applications. The first implementation of KF for navigation system is credited to Leonard McGee and Stanley Schmidt in 1961 [75]. Luinge implemented KF to measure the orientation of human body segment with accelerometers and gyroscopes [43]. Zhu uses KF for human tracking with IMU [48]. The following section outline the implementation of discrete KF discussed in chapter 2.2.2 for quaternions.

For the process model shown in equation (48), the state equations correspond to the estimation of the orientations of the IMU at $time = k + 1$based on the inputs from the gyroscopes. The rate of change of the quaternion from time k to $k + 1$ is given by Hughes [76],

$$\dot{q_k} = \frac{1}{2}\Omega_{\mathrm{k}} \cdot q_k \tag{188}$$

with

$$\Omega_{\mathrm{k}} = \begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix} \tag{189}$$

where ω_x, ω_y, and ω_z are the angular velocities for pitch, roll, and yaw in the inertial frame respectively. The state transition model sequentially integrate the orientation of the IMU with the from time k to $k+1$ becomes,

$$q_{k+1} = A_k q_k \tag{190}$$

$$A_k = \left[\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} + \frac{1}{2} \begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix} \cdot \Delta t \right] \tag{191}$$

The quaternion in the measurement model is computed as followed. The output from the magnetic sensor is first declinated by the orientation from previous iteration,

$$[h_1\ h_2\ h_3\ \ h_4] = \ q_{k-1} \otimes [0\ M_x\ M_y\ M_z] \otimes q_{k-1}^* \tag{192}$$

$$b = \begin{bmatrix} b_x & b_y & b_z \end{bmatrix} = \begin{bmatrix} {h_2}^2 + {h_3}^2 & 0 & h_4 \end{bmatrix} \tag{193}$$

$$b = b/norm(b) \tag{194}$$

The measurement model is constructed from the outputs from the accelerometer and magnetometers.

$$x_k = \begin{bmatrix} A_x\ A_y\ A_z\ M_x\ M_y\ M_z \end{bmatrix}^T \tag{195}$$

The measurement transformation from state x_k to z_k is estimated by,

$$H_k = \begin{bmatrix} -2q_{2_k} & 2q_{3_k} & -2q_{0_k} & 2q_{1_k} \\ 2q_{1_k} & 2q_{0_k} & 2q_{3_k} & 2q_{2_k} \\ 0 & -4q_{1_k} & -4q_{2_k} & 0 \\ -2b_z q_{2_k} & 2b_z q_{3_k} & -4b_x q_{2_k} - 2b_z q_{0_k} & -4b_x q_{3_k} - 2b_z q_{1_k} \\ -2b_x q_{0_k} + 2b_z q_{1_k} & -2b_x q_{2_k} + 2b_z q_{0_k} & 2b_x q_{1_k} + 2b_z q_{3_k} & -2b_x q_{0_k} + 2b_z q_{2_k} \\ 2b_x q_{2_k} & 2b_x q_{3_k} - 4b_z q_{1_k} & 2b_x q_{0_k} - 4b_z q_{2_k} & 2b_x q_{1_k} \end{bmatrix} \tag{196}$$

The process noise covariance matrix Q and the measurement noise R were determined empirically in equations (197) and (198) respectively, and the error covariance matrix P_k is initialized as a 4x4 identity matrix. Table 6-1 outlines the IMU attitude algorithm with discrete Kalman filter.

$$Q = \begin{bmatrix} 0.12 & -0.06 & 0 & -0.06 \\ -0.06 & 0.12 & -0.06 & 0 \\ 0 & -0.06 & 0.12 & -0.06 \\ -0.06 & -0.06 & -0.06 & 0.12 \end{bmatrix} \quad (197)$$

$$R = [0.001 \;\; 0.001 \;\; 0.001 \;\; 0.001] \quad (198)$$

Table 6-1 - Pseudo code for quaternionic Kalman Filter algorithm

At time = 1

Initialization:

Initialize noise covariance matrixes Q and R, error covariance matrix P_k

Initialize the initial orientation at [1 0 0 0]

At time > 1

1. Compute (H_k) ← (equation (196))
2. Compute the observation quaternion (q_{obs})

$$q_{obs} = H_k\left[A_x \, A_y \, A_z \, M_x \, M_y \, M_z\right]^T \quad (199)$$

3. Computer transition matrix (A_k) ← (equation (191))
4. Compute the prediction quaternion (q_{k+1}) ← (equation (190))
5. Compute predicted error covariance (P_{k+1}) ← (equation (65))
6. Computer Kalman gain (K) ← (equations (66-67))
7. Computer posteriori estimates $(\hat{q}_k, \hat{P}_k)$← (equations (68-69))

6.2 Extended Kalman Filter Implementation

As discussed in chapter Principle of Extended Kalman Filter, the EKF is designed to deal with the non-linear assumption of the Kalman filter estimation model. This is the most popular estimation filter within the Kalman family of the estimator as well as the most diverse. There are many different designs and implementations that optimize the algorithm to deal with different applications. Lefferts published an implementation of a quaternion EKF using multiplicative approach by assuming the 4x4 quaternion error covariance matrix must be singular [77]. Bar-Itzhack, on the other hand, used addictive approach that based on quaternion re-normalization stage [78]. In 2002, Kasdin demonstrated an alternative to EKF called two-step optimal estimator [79]. It is composed of a linear and a non-linear estimation stage, where the linear stage is the same discrete KF and the non-linear uses an optimization algorithm such as Gauss-Newton method. Goddard demonstrated the implementation of an EKF using dual quaternions approach [80].

The presented EKF implementation follows the algorithmic design by Marins [81]. The first order linearization can be achieved via Gauss Newton algorithm, and the second order can be determined with Quasi Newton method. Consider x_k from equation (195) is expressed in the Earth frame

$$x_k^E = \begin{bmatrix} 0 & 0 & 1 & M_x^E & M_y^E & M_z^E \end{bmatrix}^T \tag{200}$$

The error function can be defined as,

$$Q = \varepsilon^T \epsilon = (\overline{x_k^E} - M\overline{x_k})^T (\overline{x_k^E} - M\overline{x_k}) \tag{201}$$

$$M = \begin{bmatrix} R & 0_{3\times3} \\ 0_{3\times3} & R \end{bmatrix} \tag{202}$$

$$\overline{x_k^E} = \begin{bmatrix} A_x & A_y & A_z \\ M_x & M_y & M_z \end{bmatrix}, \qquad \overline{x_k} = \begin{bmatrix} 0 & 0 & 1 \\ M_x^E & M_y^E & M_z^E \end{bmatrix} \tag{203}$$

where R is the rotation matrix from $time = k - 1$ given by equation (19).

The first order derivative of the error function is Jacobian matrix of Q

$$\mathrm{J} = -\left[\frac{\partial M}{\partial q_{0_k}} x_k \quad \frac{\partial M}{\partial q_{1_k}} x_k \quad \frac{\partial M}{\partial q_{2_k}} x_k \quad \frac{\partial M}{\partial q_{3_k}} x_k\right] \tag{204}$$

$$J = \begin{bmatrix} A_x q_{2_{k-1}} + A_y q_{3_{k-1}} + A_z q_{4_{k-1}} & -A_x q_{1_{k-1}} + A_y q_{0_{k-1}} + A_z q_{3_{k-1}} \\ A_x q_{1_{k-1}} - A_y q_{0_{k-1}} - A_z q_{3_{k-1}} & A_x q_{0_{k-1}} + A_y q_{1_{k-1}} + A_z q_{3_{k-1}} \\ A_x q_{2_{k-1}} + A_y q_{3_{k-1}} - A_z q_{0_{k-1}} & -A_x q_{3_{k-1}} + A_y q_{2_{k-1}} - A_z q_{1_{k-1}} \\ M_x q_{1_{k-1}} + M_y q_{2_{k-1}} + M_z q_{3_{k-1}} & -M_x q_{2_{k-1}} + M_y q_{1_{k-1}} + M_z q_{0_{k-1}} \\ M_x q_{2_{k-1}} - M_y q_{1_{k-1}} - M_z q_{0_{k-1}} & M_x q_{1_{k-1}} + M_y q_{2_{k-1}} + M_z q_{3_{k-1}} \\ M_x q_{3_{k-1}} + M_y q_{0_{k-1}} - M_z q_{1_{k-1}} & -M_x q_{0_{k-1}} + M_y q_{3_{k-1}} - M_z q_{2_{k-1}} \end{bmatrix}$$

$$\begin{bmatrix} -A_x q_{2_{k-1}} - A_y q_{3_{k-1}} + A_z q_{0_{k-1}} & A_x q_{3_{k-1}} - A_y q_{2_{k-1}} + A_z q_{1_{k-1}} \\ A_x q_{3_{k-1}} - A_y q_{2_{k-1}} + A_z q_{1_{k-1}} & A_x q_{2_{k-1}} + A_y q_{3_{k-1}} + A_z q_{0_{k-1}} \\ A_x q_{0_{k-1}} + A_y q_{1_{k-1}} + A_z q_{2_{k-1}} & -A_x q_{1_{k-1}} + A_y q_{0_{k-1}} + A_z q_{3_{k-1}} \\ -M_x q_{2_{k-1}} - M_y q_{3_{k-1}} + M_z q_{0_{k-1}} & M_x q_{3_{k-1}} - M_y q_{2_{k-1}} + M_z q_{1_{k-1}} \\ M_x q_{3_{k-1}} - M_y q_{2_{k-1}} + M_z q_{1_{k-1}} & M_x q_{2_{k-1}} + M_y q_{3_{k-1}} + M_z q_{0_{k-1}} \\ M_x q_{0_{k-1}} + M_y q_{1_{k-1}} + M_z q_{2_{k-1}} & -M_x q_{1_{k-1}} + M_y q_{0_{k-1}} + M_z q_{3_{k-1}} \end{bmatrix} \tag{205}$$

The iterative Gauss Newton step is defined as,

$$qn_k = {q_{k-1}}^T - \left(\frac{1}{J^T J}\right) J^T \left(\overline{x_k^E} - M(\overline{x_k})\right) \tag{206}$$

The covariance matrixes Q and R, and the error covariance P_k were defined the same as the KF. Table 6-2 outlines the IMU attitude algorithm with EKF.

Table 6-2 - Pseudo code for quaternionic Extended Kalman Filter algorithm

At time = 1

Initialization:

Initialize noise covariance matrixes Q and R, error covariance matrix P_k

Initialize the initial orientation at [1 0 0 0]

At time > 1

For steps = 1→ N

1. Compute (J) ← (equation (205))
2. Compute quaternion via Gauss Newton step (qn_k) ←(equation (206))

Return qn_k

End

$qn_k = q_{obs}$

3. Computer transition matrix (A_k) ← (equation (191))
4. Compute the prediction quaternion (q_{k+1}) ← (equation (190))
5. Compute predicted error covariance (P_{k+1}) ← (equation (83))
6. Computer Kalman gain (K) ← (equations (84-85))
7. Computer posteriori estimates $(\hat{q}_k, \hat{P}_k)$← (equations (86-87))

6.3 Unscented Kalman Filter

Unscented Kalman filter is designed to restore the normality of the system process that was lost in the linearization method in EKF as discussed in chapter 2.2.5. One of the issues with UKF is the sigma points calculation in equations (102-105). The square root function of the matrix is typically achieved by lower triangle Cholesky decomposition. However, Cholesky's method require the matrix to be positive definite, which is not always true depending on the choice of α, κ and the dimensionality L in equation (103). Cheon simulated a quaternionic UKF that calculates barycentric mean of the sigma point quaternions by renormalization. However, the matrix square root issue was never discussed.

In 2003, Crassidis demonstrated a practical implementation of quaternionic UKF by converting the quaternions into modified Rodrigues parameters (MRP) during the sigma point calculation [82]. This method raises another issue, which is the statistical assumption that the transformation between quaternion and modified Rodrigues parameters. Upon transformation, the sigma points of the quaternion, which carries the statistical information, are projected from its hyper-dimensional manifold onto the 3-dimensional (3D) space. After the estimation and filtering process, the 3D modified Rodrigues parameters are projected back to quaternion. However, the distribution cannot be fully restored the statistical information as a part of the information is lost during projection.

Consider this problem in one less dimension with a generic 3D distribution in Figure 6-1, the projection of 3D distribution onto 2D space is equivalent of taking a 'slice' of the distribution. The projection will be similar to a generic Gaussian curve. However, the Gaussian curve cannot return to a 3D distribution without making statistical

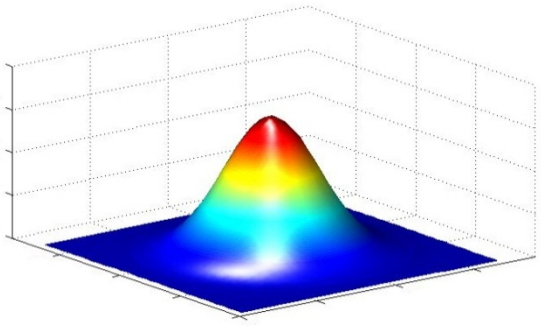

Figure 6-1- Generic 3D distribution

assumptions on the true mean and variance of the distribution. This is demonstrated in a simulation study in Appendix C. For biomedical tracking application, Harada used quaternionic UKF with IMU as a portable orientation tracking device [83].

6.4 Complementary filter implementation

Complementary filter one of the most used implementation for attitude heading references system (AHRS) for standalone embedded system such as unmanned vehicle because of its compact algorithm design. Mahony demonstrated various implementations of the complementary filters [72]. Sebastian used a gradient descent method to estimate the drift of the gyroscope as part of the filter [84].

The following implementation follows Mahony's complementary design. The data from the magnetometers is first processed by equations (192-194). The quaternion from prior state is transformed to the gravity and magnetic fields by,

$$v_a = \begin{bmatrix} 2q_{1_k}q_{3_k} - q_{0_k}q_{2_k} \\ 2q_{0_k}q_{1_k} + q_{2_k}q_{3_k} \\ {q_{0_k}}^2 - {q_{1_k}}^2 - {q_{2_k}}^2 + {q_{3_k}}^2 \end{bmatrix} \tag{207}$$

$$v_m = \begin{bmatrix} 2b_x\left(0.5 - {q_{2_k}}^2 - {q_{3_k}}^2\right) + 2b_z\left(q_{1_k}q_{3_k} - q_{0_k}q_{2_k}\right) \\ 2b_x\left(q_{1_k}q_{2_k} + q_{0_k}q_{3_k}\right) + 2b_z\left(q_{0_k}q_{1_k} + q_{2_k}q_{3_k}\right) \\ 2b_x\left(q_{0_k}q_{2_k} - q_{1_k}q_{3_k}\right) + 2b_z\left(0.5 - {q_{1_k}}^2 - {q_{2_k}}^2\right) \end{bmatrix} \tag{208}$$

The error function is defined by,

$$E_{mes} = ([A_x \quad A_y \quad A_z] \times v_a + [M_x \quad M_y \quad M_z] \times v_m) \tag{209}$$

The integral drift error is calculated with

$$E_{drift_k} = E_{drift_{k-1}} + E_{mes}\Delta t \tag{210}$$

The corrective feedback term is applied,

$$[\omega_x \quad \omega_y \quad \omega_z] = [\omega_x \quad \omega_y \quad \omega_z] + K_p E_{mes} + K_i E_{drift_k} \tag{211}$$

where K_p and K_i are error gains. The prediction is then performed with equation (190) and (191).

6.5 Sequential Monte Carlo Methods Implementation

The fundamental problem with the Kalman estimation family for quaternion is the construction of the error covariance matrix. The components of the quaternion vary with each other as a function to maintain the unit norm constraint. Many implementations and variations optimize the error covariance via projections or various ad hoc techniques such as modifying and restoring the quaternion to maintain the unit norm property. The current implementation of EKF obtains these matrixes empirically through trial and error.

Sequential Monte Carlo method or particle filter (PF) does not use the error covariance directly in its calculation, instead, it approximate the distribution through a finite set of independent and identically distributed samples. The challenge is to generate a statistical geometry for the quaternion. Yang [85] and Cheng [86] implemented the quaternionic PF using the similar technique from Crassidia's UKF by projecting the quaternion with MPR to perform statistical analysis. IN 2010, Yang introduced the Gaussian sum PF for quaternion, which uses expectation maximization

method to approximate the Gaussian mixture [87]. Carmi also implemented a quaternionic PF by using genetic algorithm to approximate the likelihood function [88].

The PF implementation in this research utilized high dimensional directional statistical method to generate and examine the quaternions. The following sections review and discuss the principles and implementations of these statistical geometries.

6.5.1 Directional Statistic

Directional statistic is a special subset of statistic that primarily deals with directions, axes and rotations. The problem with these types of geometry is the mathematical singularity point or the switching of polarities, which cannot be processed by traditional statistical calculations without special rules and exceptions. For example, the mean of the rotations at 10 and 350 degrees yield 180 degrees instead of 0.

6.5.2 Stiefel Manifold

The topological space of a collection of p-dimensional orthonormal vectors in N-dimensional space is considered to be the Stiefel Manifold (V_p), which is defined as [89],

$$V_p(\mathbb{R}^N) = \{\boldsymbol{A} \in \mathbb{R}^{N\times p} : \boldsymbol{A}^*\boldsymbol{A} = 1\} \tag{212}$$

where $\mathbb{R}^N$ can be any inner product space.

For quaternion, where $p = 4$ and $N = 3$, satisfies such condition and forms a unique case on the manifold,

$$V_p(\mathbb{R}^N) = \{q \in \mathbb{R}^{N\times p} : q^* \otimes q = I_p\} \tag{213}$$

where $I_p = [1\ 0\ 0\ 0]$.

Statistical distributions residing on the Stiefel manifold generally includes any arbitrary dimensional objects in any dimensional space. In general, any distribution satisfying the condition of $p < N$ can be used as a quaternionic distribution. Two of these

distributions were examined in this research, which are the von Mises-Fisher distribution and the Bingham distribution. The reason of these distributions is interesting is that von-Mises Fisher is a uniformly distribution in 4D space while Bingham distribution allows more complex and non-uniform statistical geometries [90]. The formulation of these distributions is discussed below.

6.5.3 von Mises- Fisher Distribution

The von Mises Fisher (vMF) distribution is a generalized spherical distribution of p-dimensional object in $p-1$ dimensional space. The probability density function (pdf) of a generalized von Mises Fisher distribution of p-dimensional object is given as [91],

$$f_{vMF}(X;F) = \frac{1}{a(F)} e^{tr(FX^T)} \tag{214}$$

where X is a $p \times N$ matrix of orthogonal unit vectors, F is a $p \times N$ parameters matrix, and $1/a(F)$ is the normalizing constant, which can be expressed by a confluent hypergeometric limit function,

$$a(F) = {}_0F_1\left(\frac{N}{2}, \frac{FF^T}{4}\right) \tag{215}$$

$$= \boldsymbol{I_v}(F) \frac{\Gamma\left(\frac{-N}{2}+1\right)}{\left(\frac{F}{2}\right)^{-\frac{N}{2}}} \tag{216}$$

where $\boldsymbol{I_v}$ is the Bessel function of the first kind, Γ is the gamma function.

The distribution applied to quaternion with $p = 4$ becomes [92],

$$f_{vMF}(\boldsymbol{x};\mu,\kappa) = C_4(\kappa) e^{(\kappa\mu^T\boldsymbol{x})} \tag{217}$$

$$C_4 = \frac{\kappa}{2\pi^2 \boldsymbol{I_v}(\kappa)} \tag{218}$$

where $\boldsymbol{x}$ is a random quaternion, μ is the mean vector, and κ is the dispersion factor.

Direct statistical inference with the von Mises-Fisher distribution is often impractical. However indirect approaches using expectation maximization methods

[93,94,95] can be used to sample the distribution indirectly. An efficient sampling method is proposed by Wood in 1994 [96]. Wood's algorithm is based on Ulrich's simulation proposal for m-sphere [97]. Instead of trying to generate samples from the distribution, the algorithm simulates random samples that have the statistical properties of the real distribution. Ulrichs's theorem postulated that a unit p-vector X has a von Mises-Fisher distribution with mean direction at $X_0 = [1\ 0\ 0\ 0]$ if and only if

$$X^T = \left(V\sqrt{1-W^2}, W\right) \tag{219}$$

where V is a uniformly distributed unit (p-1) vector and W is the scalar random variable ranging from -1 to 1. The von Mises-Fisher simulation comes down to determining an efficient method to simulate W, which is calculated with Ulrich's proposal by using an envelope proportional to the density along with beta random variables.

$$e(x,b) = d_{m,b}^{-1}(1-x^2)^{\frac{(m-3)}{2}}(1+b-(1-b)x)^{-(m-1)} \tag{220}$$

$$d_{m,b}^{-1} = \left(\Gamma\left(\frac{(\mathrm{m}-1)}{2}\right)\right)^2 b^{-\frac{m-1}{2}} \Big/ 2\Gamma(\mathrm{m}-1) \tag{221}$$

$$b = \frac{-2F + \sqrt{4F^2 + (m-1)^2}}{m-1} \tag{222}$$

$$Z \sim \beta[-(m-1)/2, (m-1)/2] \tag{223}$$

$$W = {1-(1+b)Z} \Big/ {1-(1-b)Z} \tag{224}$$

The simulation algorithm is shown in Table 6-3. Figure 6-2 shows the output from the von Mises-Fisher simulation with different level of dispersion parameters at the mean direction of [1 0 0 0]. As the dispersion factor increases, the concentration of the sample increases. Since the distributions below is projected to the 3-sphere with an identity matrix, the figures below do not represent the full distribution but an instance of it.

Table 6-3 - Pseudo code for von Mises distribution simulation

Input: μ (mean vector),
κ (dispersion factor),
N (number of samples/particles)

1. $b = -\kappa + \sqrt{\kappa^2 + 1}$
2. $x_0 = \frac{1-b}{1+b}$
3. $c = \kappa(x_0) + 2\log(1 - x_0 x_0)$
4. for n = 1→N
5. while t ≤ u
6. while s ≤ 1
7. $uu \sim \prod(-1,1)$, $vv \sim \prod(0,1)$
8. $s = uu + vv$

   End
9. $z = \frac{1}{2} + uu * vv * \frac{\sqrt{1-s}}{s}$
10. $u \sim \prod(0,1)$
11. $w = \frac{1-z(1+b)}{1-z(1-b)}$
12. $t = \kappa(w) + 2\log(1 - x_0 w) - c$

    End
13. $\theta \sim \prod(0,2\pi)$, $u \sim \prod(-1,1)$
14. $v = \sqrt{1 - uu}$
15. $rand3DVec = [v * \cos(\theta) \quad v * \sin(\theta) \quad u]$
16. $q_r = w$
17. $q_{x,y,z} = \sqrt{1 - w^2} * rand3DVec$
18. $q = [q_r \; q_x \; q_y \; q_z]$
19. $q_{vMF}(n) = q \otimes \mu$

    End
20. Return q_{vMF}

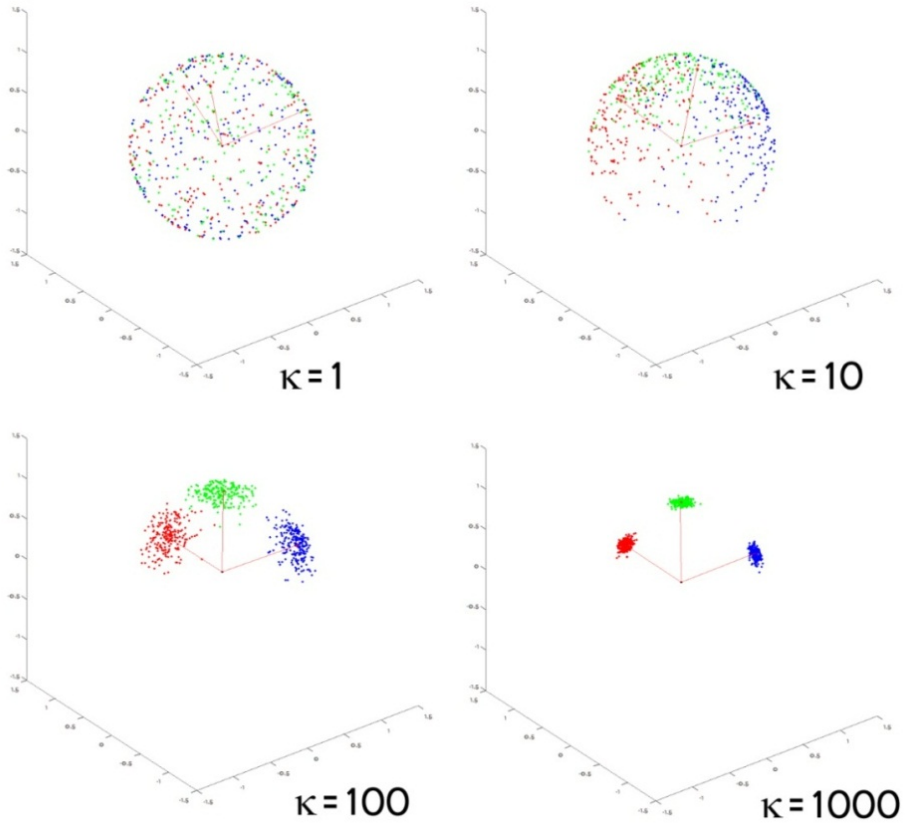

Figure 6-2 – Randomly sampled quaternions with von Mises-Fisher distributions with different dispersion factor. The samples are projected to 3-sphere with identity matrix. The red lines indicate the mean direction of the samples.

6.5.4 Bingham Distribution

The von Mises-Fisher distribution is a subclass of a generic higher dimensional distribution known as Bingham distribution. The von Mises-Fisher assumes the samples are uniformly distributed around the mean direction of the rotation manifold. Bingham distribution is a statistical distribution for hyper-dimensional object that does not assume rotational symmetry and uniformity. The distribution is extremely flexible that can represent even elliptic or girdle distribution geometry. The probability density for the Bingham distribution is defined as

$$f_B(\pm q; K) = {}_1F_1\left(\frac{1}{2}, \frac{p}{2}, K\right)^{-1} e^{q^T U K U^T q} \tag{225}$$

$${}_1F_1\left(\frac{1}{2}, \frac{p}{2}, K\right) = \sum_{n=0}^{\infty} \frac{1/2^{(n)}}{p/2^{(n)}} \frac{K^n}{n!} \tag{226}$$

where q is the quaternion describing the orientation, ${}_1F_1\left(\frac{1}{2}, \frac{p}{2}, K\right)^{-1}$ is Kummer's function of the first kind as normalizing constant, U is an orthogonal matrix describing the orientation of the distribution, and $\boldsymbol{K}$ is diagonal matrix that describes the dispersion axes of the distribution defined as,

$$\boldsymbol{K} = \begin{bmatrix} \kappa_s & 0 & 0 & 0 \\ 0 & \kappa_1 & 0 & 0 \\ 0 & 0 & \kappa_2 & 0 \\ 0 & 0 & 0 & \kappa_3 \end{bmatrix} \tag{227}$$

Similar to the von Mises-Fisher distributions, Bingham distribution cannot be sampled directly, and indirect simulation method is used. Hoff demonstrated the generation of random samples from the Bingham distribution using Gibbs sampling algorithm [98]. Glove applied Metropolis-Hasting algorithm for sampling random quaternions from the Bingham density [99].

In this research, a rejection sampling algorithm was designed to create a set of random simulation samples from the Bingham density. Rejection criterion is based on the maximum and minimum acceptance densities. The algorithm design is shown in the Table 6-4. As shown in the table, there are two methods to initialize the samples for the rejection algorithm, which are random hypersphere simulation shown in step 2 to 10, and the von Mises-Fisher simulation method on step 11. According to equation (225), Bingham density models an antipodal symmetric distribution. Hence, rejection sampling using random hyperspheres will result with bimodally distributed sample as the equation accepts the quaternions and its complex conjugates that fall within the acceptance range. This can adversely affect the expectation's direction of the random samples as it becomes unpredictable when projecting back to 3D. This is highly undesirable for tracking applications.

A secondary proposal is to use the samples created from the von Mises-Fisher simulation with small dispersion factor to initialize the sampling with a large particles spread. This will remove the possibility of sampling the complex conjugate of the quaternion. In addition, initializing with von Mises-Fisher density reduces the time to generate samples significantly as it restricts the seeking space of the samples. However, since the samples generated from this method eliminate the antipodal properties of the distribution, it cannot be considered as the Bingham distribution. This is referred as the non-uniform (NU) distribution and density in the following sections. Figure 6-3 shows the output from the simulation with different dispersion matrix ($\boldsymbol{K}$) at the mean direction of [1 0 0 0].

Table 6-4 - Pseudo code for Non-Uniform distribution simulation

Input: μ (mean vector),
$\boldsymbol{K}$ (dispersion matrix),
U (Orthogonal matrix of the distribution's symmetry axes)
f_{max}, f_{min} (Maximum and minimum accepting densities)
N (number of samples/particles)

1. $x_i = \infty, i = 1,2,3,4$
2. while n<N
3. If ${x_1}^2 + {x_2}^2 \geq 1$ or${x_3}^2 + {x_4}^2 \geq 1$
4. $x_i \sim \prod(-1,1)\,, i = 1,2,3,4$
5. else
6. $t = \frac{1-{x_1}^2+{x_2}^2}{{x_3}^2+{x_4}^2}$
7. $q_1 = x_4 \times t, \quad q_2 = x_1, \quad q_3 = x_2, \quad q_4 = x_3 \times t$
   end (go to 12.)
   -- OR --
8. Use von Mises-Fisher simulation in
9. Table 6-3
10. $f = {}_1F_1\left(\frac{1}{2}, \frac{p}{2}, q\right)^{-1} e^{\boldsymbol{q^T UKUq}} \propto e^{\boldsymbol{q^T UKUq}}$
11. If $f > f_{min}$ & $f < f_{max}$
12. $q_{NU}(n) = q \otimes \mu$
13. $n = n + 1$
    end
    end
    Return q_{NU}

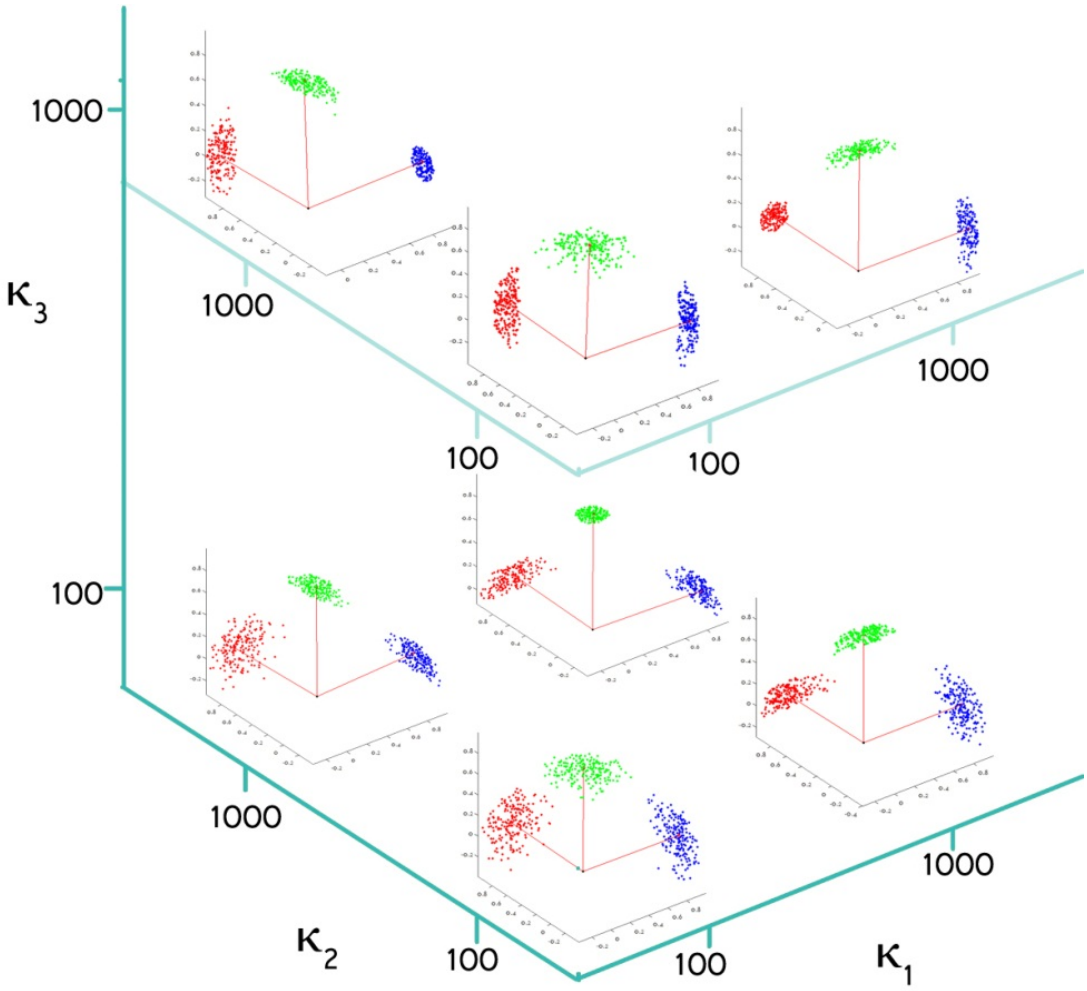

Figure 6-3 – Randomly sampled quaternions with non-uniform distributions with different density proportion in *K*. The samples are projected to 3-sphere with identity matrix. The red lines indicate the mean direction of the samples.

6.5.5 Sequential Monte Carlo Methods with von Mises-Fisher Density

As discussed in chapter 2.2.6, there are essentially four stages in particle filtering (PF), which are particles generation, states evolution, particles evaluation and particle maintenance. In this section, the implementation of PF for tracking application based on the von Mises-Fisher density is examined. The overall block diagram of the algorithm is shown in Figure 6-4.

One of the challenges in formulating PF is to tie the particles generation, particles evaluation and particles maintenance together such that the particles can be weighed correctly and to produce the optimal importance density that reflects the state of the estimation. The following method establishes a correlation between the uncertainties of the random particles and the dispersion factors such that particles can be generated and evaluated based on the posterior density.

A set of N particles samples is simulated at different dispersion factor at the mean direction [1 0 0 0]. The rotational uncertainty of these particles is determined by the root sum squared of the minimum angle between two hyper-complex vectors.

$$\delta_i = \sqrt{\sum_i^N \left(2 * \operatorname{acos}\left(\left|q_{x,i}^{\kappa} \cdot q_0\right|\right)\right)^2}$$
$$q_0 = [1\ 0\ 0\ 0]\,, \qquad q_{x,i}^{\kappa} \sim \mathbf{vMF}(\kappa)\,, \qquad i = 1 \to N, \kappa = \kappa_{min} \to \kappa_{max} \tag{228}$$

The relationship between δ_i and κ is realized with least squared approximation of the two datasets with the function in equation (229), and Figure 6-5.

$$\kappa(\delta, \boldsymbol{x}) = ae^{-b(\delta)} + ce^{-d(\delta)}, \qquad \boldsymbol{x} = [a\ b\ c\ d] \tag{229}$$

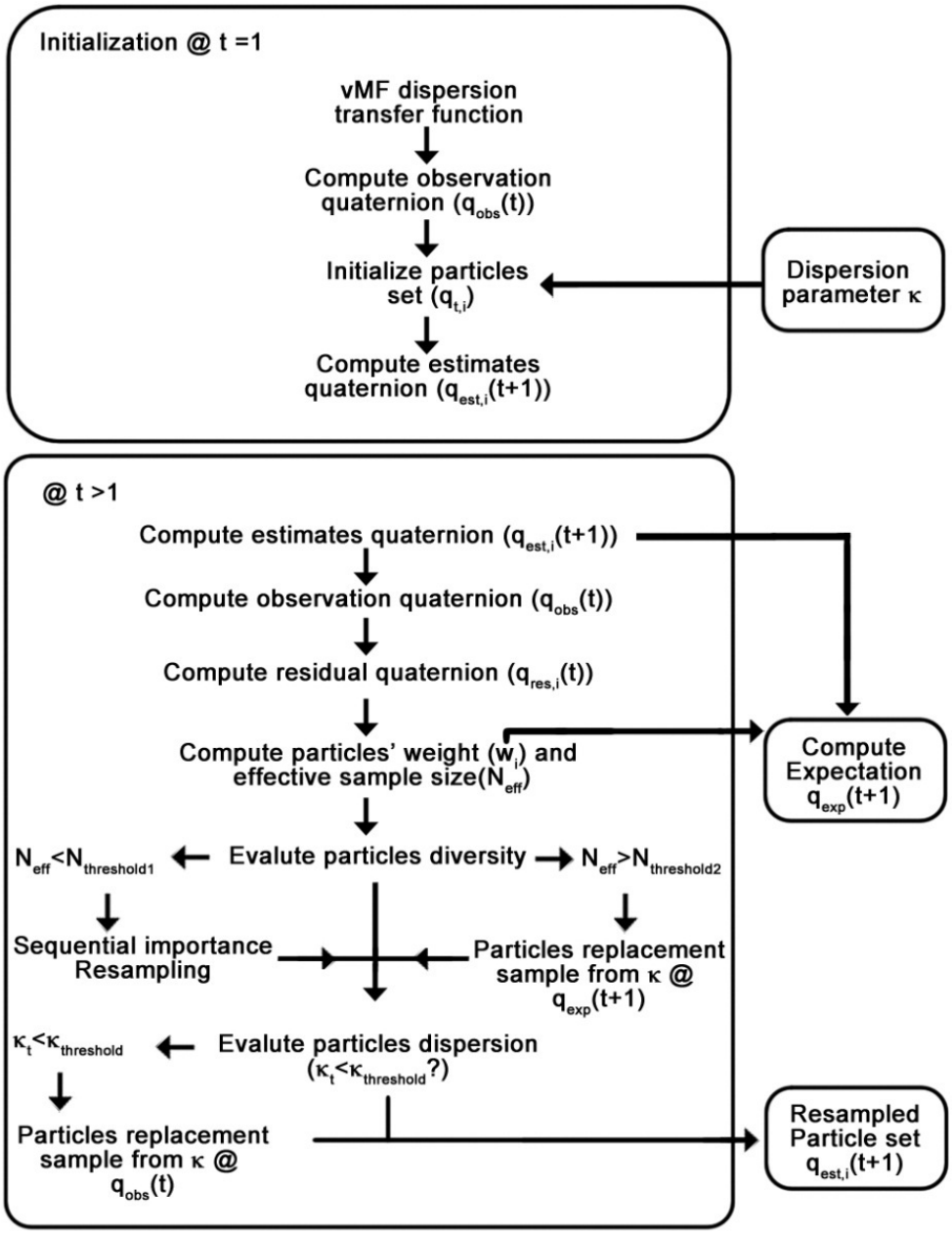

Figure 6-4 – Functional block diagram of the PF algorithm with von Mises-Fisher density

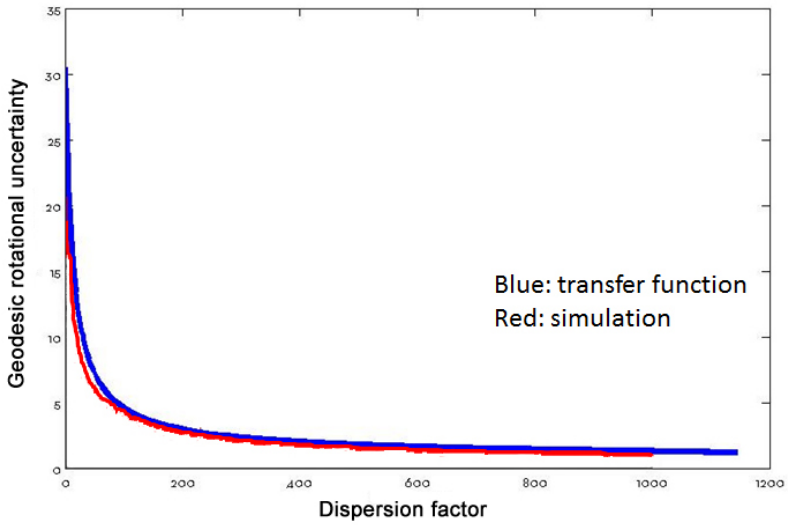

Figure 6-5 – Estimation function between particles uncertainty and dispersion factor

During the initialization procedure, the observation quaternion is initialized with the Gauss Newton method demonstrated in equations (200-206). A set of N particles are computed based on the Wood's simulation in Table 6-3 with an arbitrary dispersion factor. The initial particles estimates ($q_{est,i}(t)$) become,

$$q_{est,i}(t) \sim \mathbf{vMF}(\kappa) \otimes q_{obs}(t), i = 1 \dots N \tag{230}$$

All the particles are assigned with equal weights during the initialization period. The weights and dispersion factor are updated accordingly in the subsequent cycle by the posterior filtering density. The algorithm then computes the particles estimates for the next cycle at $time = t + 1$,

$$q_{est,i}(t+1) = q^{\kappa}_{est,i}(t) + 0.5\left(q^{\kappa}_{est,i}(t) \otimes \begin{bmatrix} 0 & \omega_x & \omega_y & \omega_z \end{bmatrix}\right)\Delta t, \qquad i = 1 \dots N \tag{231}$$

where ω are the angular rate measured at time t, and Δt is the sampling period.

After the initialization, the recursive portion of the algorithm begins by first determining the observation quaternion $q_{obs}(t)$ at the current time, and the estimates of

the next cycle are computed with equation (231). Particles evaluation depends on the hypothesis of the optimal importance density. Since there are two parameters in the von Mises-Fisher density, the ideal choice for the orientation tracking is the residual density defined in equation (232). This is because the optimal mean direction of the residual particles is always [1 0 0 0] instead of an arbitrary quaternion.

$$q_{res,i}(t) = q_{est,i}(t) \otimes conj\big(q_{obs}(t)\big), \qquad i = 1 \dots N \tag{232}$$

The second parameter, the dispersion factor, is approximated sequentially with following method. The weight of the particles must first be determined. The rotational disparity between the optimal residual quaternion and the residual particles is given by,

$$\vartheta_{res,i} = 2\cos(q_{res,i}(t) \cdot q_0), \qquad i = 1 \dots N \tag{233}$$

The rotational difference is then used to determine the importance weights of the particles estimates, where less rotational discrepancy receives higher weight and vice versa.

$$w_i = \frac{1/\delta_{res,i}}{\sum_i^N (1/\delta_{res,i})}, \qquad i = 1 \dots N \tag{234}$$

The posterior dispersion parameter is updated with,

$$\kappa\left(\sqrt{\sum_i^N {\vartheta_{res,i}}^2}, \boldsymbol{x}\right) = ae^{-b\left(\sqrt{\sum_i^N {\vartheta_{res,i}}^2}\right)} + ce^{-d\left(\sqrt{\sum_i^N {\vartheta_{res,i}}^2}\right)}, \tag{235}$$
$$\boldsymbol{x} = [a\ b\ c\ d],\ i = 1 \dots N$$

The expectation of the filtered quaternion is computed with the particles estimates and their weights. This is accomplished by computing the weighted spherical averages of the particles in a successive double loop shown in Table 6-5.

Table 6-5 - Pseudo code for expectation calculation for quaternion

Input: $q_{est,i}, i = 1 \rightarrow N$ (Estimation data),
$w_i, i = x, y, z$ (Data weights),
N (number of particles)

1. for x = 1→ log(N)/log(2)
2. for k = 1 →(**size**($q_{est,i}$))/2
3. $w_n = w_{2k-1}/(w_{2k-1} + w_{2k})$
4. $\theta = \text{acos}(q_{est,2k-1} \cdot q_{est,2k})$
5. $q_{V,k} = q_{est,2k-1}\left(\frac{\sin((1-w_n)\theta)}{sin\theta}\right) + q_{est,2k}\left(\frac{\sin((w_n)\theta)}{sin\theta}\right)$
6. $w_{V,k} = w_{2k-1}(w_n) + w_{2k}(1 - w_n)$

end
7. $q_{V,k} \rightarrow q_{est,i}, i = 1 \rightarrow k$
8. $w_{V,k} \rightarrow w_i, i = 1 \rightarrow k$

end
Return q_V

There are various techniques to compute spherical averaging [100]. Buss demonstrated spherical averaging with exponential mapping technique [101]. Clark and Thompson used stereographic projection for averaging [102].

The presented algorithm uses spherical linear interpolation (SLERP) [103] to interpolate the rotation between two quaternion. SLERP computes the intermediate rotation of two orientation inputs and the weight parameter ranging from 0 to 1. The weight parameter determines the influence of each of the rotational inputs to the output. A weight of 0.5 is equal to calculating the mean rotation of the two inputs. Since all the particles are weighted differently, the weights are first normalized between two examining particles (Step 3). The weight of the output is computed by the ratio of the normalized weight (Step 6). In each iteration, the algorithm interpolates a new generation of particles and weights that is half the size of the input (steps 2 to 8) until

there is only one particle left. This algorithm is limited to sample size of 2^n. Zero padding is for this algorithm necessary if other sample size is used.

Particle maintenance is to ensure the effectiveness of the particle estimates for statistical inference and to avoid degeneracy. The first step is to determine the effective sample size N_{eff} described in equation (156) in chapter 2.2.6. In the current implementation of the PF, the particle maintenance is accomplished by two scenarios.

In the first scenario, two thresholds are set by the user. The first threshold (N_{th1}) determines whether the particle samples require importance resampling. If the N_{eff} is smaller than N_{th1}, importance resampling is performed. There are three importance resampling methods examined in this research, which are deterministic [104], residual [69], and auxiliary [105] as shown in Appendix D.

The second threshold (N_{th2}) is used to enrich the particles' diversity. This threshold is added in addition to the original because instead of the importance weights' degeneracy; the importance density becomes highly concentrated from the resampling where a large population of the particles estimates becomes identical. If N_{eff} is larger than N_{th2}, the particles are replaced by the new particles that are sampled from the posterior density in equation (235) and the expectation quaternion from the algorithm in Table 6-5.

$$q_{rs,i}(t) \sim \mathbf{vMF}\left(\kappa\left(\sqrt{\sum_{i}^{N} {\delta_{res,i}}^2}, \boldsymbol{x}\right)\right) \otimes \left(q_{exp}(t+1)\right) \qquad i = 1 \dots N \tag{236}$$

The weight is also updated based on the resampled particles. This step increases the diversity of the particles while maintaining the statistical properties of the particles.

The second scenario takes into account of the hardware system of the IMU. Several cases were identified that can cause temporary interruption or interference to the system. For instance, external magnetic field such as laptop batteries or speakers can disturb the magnetometer's measurement. Motion that causes temporary sensing signal saturation can lead to incorrect estimations. In addition, the antenna on the IMU may be obscured by the user. This causes a transmission lag as shown in Figure 6-6, which is highly undesirable for rate-dependent estimation.

These events may lead to a significant error in the posterior density approximation as the prediction and observation are misrepresented. The dispersion factor, which is based on the state of the residual particles, will decrease to increase the spread of the particles. It is possible for the filter to destabilize if the dispersion factor is low enough that the particles become completely random hyperspheres. Since the effectiveness of the particles is monitored by normalized importance weight, it cannot directly observe if the residual particles are drifting away from the optimal direction. The most direct method is to monitor the posterior density, which infers the uncertainty state of the residual particles. Therefore, a dispersion threshold (κ_{TH}) is set up to prevent excessive dispersion. If the dispersion calculated in (235) drops below κ_{TH}, the particles are reset by sampling a new set particles at $\left(q_{obs}(t)\right)$ with the initial dispersion factor used at $time = 1$.

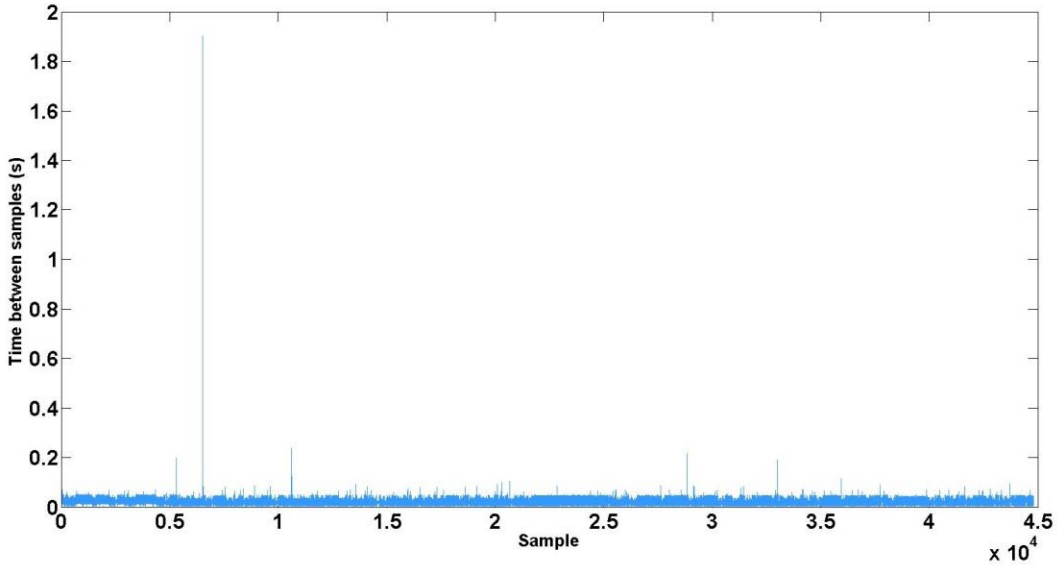

Figure 6-6 - Receiving time lag between samples

6.5.6 Sequential Monte Carlo Methods with non-Uniform Density

Since the sampling method between von Mises-Fisher and non-uniform densities are substantially different from one another, the set up for the PF, and the particles maintenance procedures need to be re-designed. In non-uniform sampling method described in chapter 6.5.4, the shape of the density is governed by the dispersion shape matrix K, defined in equation (227); and the amount of dispersion is controlled by the maximum and minimum acceptance boundaries (f_{max} and f_{min}). The overall function block for PF with NU density is shown in Figure 6-7. The particles are initialized with sampling with the rejection sampling method outlined in Table 6-4. The initial particles estimates ($q_{est,i}(t)$) become,

$$q_{est,i}(t) \sim \mathbf{NU}([f_{max}, f_{min}], \boldsymbol{K}) \otimes q_{obs}(t), \qquad i = 1 \dots N \tag{237}$$

These particles are weighted equally. The particles estimates are computed with equation (231). The uncertainty states of the particles are determined with equation (232-234). The effective sample size is calculated with equation (156) in chapter 2.2.6.

Sequential importance resampling is performed in the same manner as the PF with von Mises-Fisher density method. However, to enrich the particle diversity after the effective particle size N_{eff} exceeds N_{th2}, replacement particles must be sampled from the density that describes the current particle state. This is achieved by first determining the density of the residual quaternions,

$$f_{res,i} = {}_1F_1\left(\frac{1}{2}, \frac{p}{2}, q_{res,i}(t)\right)^{-1} e^{q_{res,i}(t)^T \boldsymbol{UKU} q_{res,i}(t)}, \qquad i = 1{:}N$$
$$\propto e^{q_{res,i}(t)^T \boldsymbol{UKU} q_{res,i}(t)} \tag{238}$$

The maximum and minimum residual densities are used as the new acceptances boundaries, where the replacement particles are drawn from.

$$q_{rs,i}(t) \sim \mathbf{NU}([f_{max}, f_{min}], \boldsymbol{K}) \otimes \left(q_{exp}(t+1)\right) \qquad i = 1 \ldots N \tag{239}$$

The state of the residual particles direction is monitored by two pre-determined densities boundaries (f_1 & $f_2, f_1 < f_2$) to prevent particles divergence from fault inputs. If $[f_{max}, f_{min}]$ is not within the density bounds defined by f_1 & f_2, the particles are reset by sampling a new set particles at $\left(q_{obs}(t)\right)$ with the initial boundaries at time = 1.

6.5.7 Sequential Monte Carlo Methods with Bias compensation

Gyroscopes drifting bias is one of the major causes in attitude estimation error. The PF discussed in previous sections can only observe and correct drifting error via monitoring the direction of the residual density and correct the particles only if they jeopardize the stability of the filter. Hence, the bias correction developed by Mohany [72] is used in conjunction to produce the particle estimates. This can enhance the stability and accuracy of the filter as it reduces the occurrence of particles replacement caused by large prediction and update error; where the particles are reset.

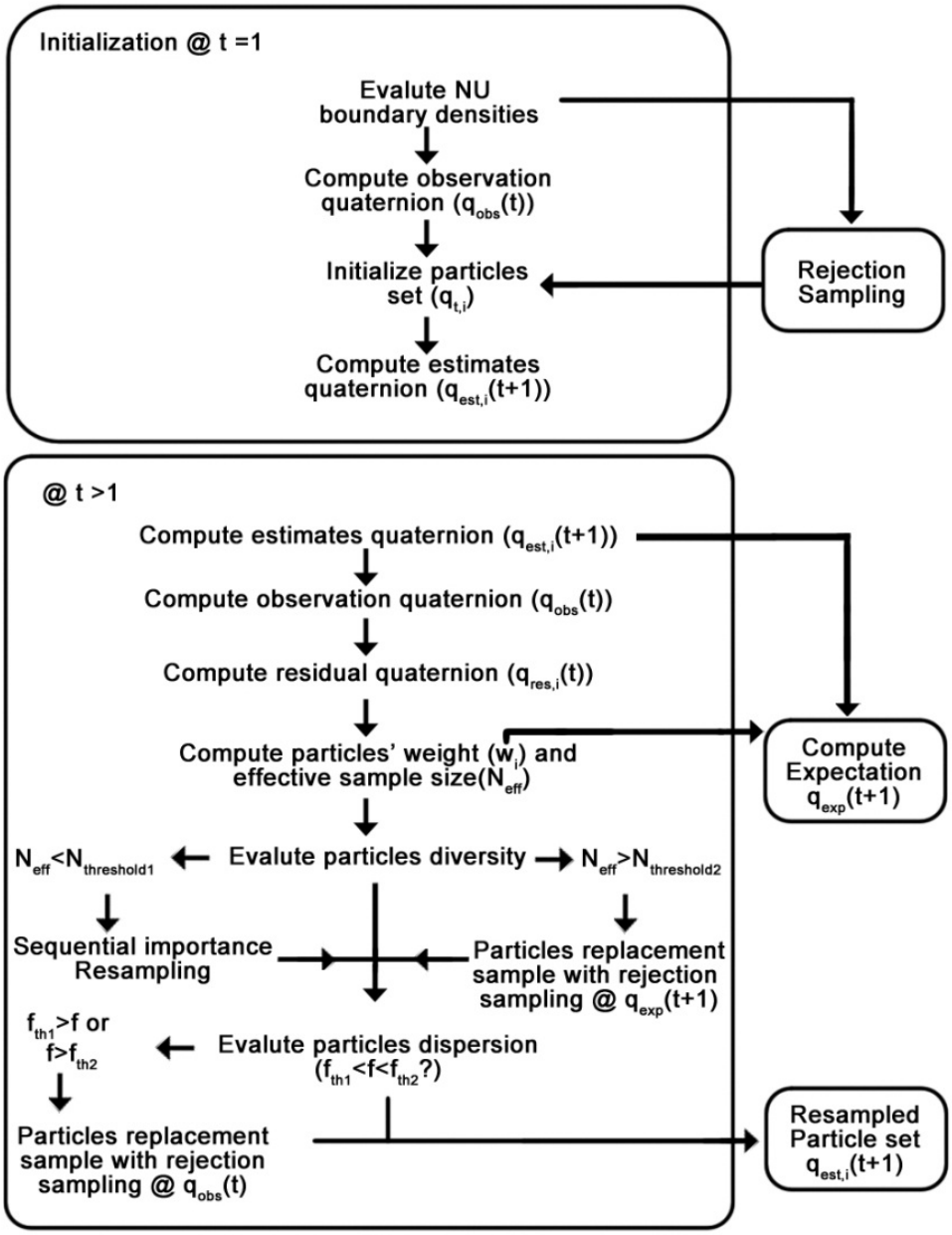

Figure 6-7 - Functional block diagram of the PF algorithm with non-uniform density

7. Experimental Results

Three experiments were designed to test the IMU system and the algorithm design. The first experiment focuses on the validation of the algorithms by testing it with synthetic signal and compare against benchmarks algorithms such as extended Kalman filter (EKF) and complementary filter (CF). The second experiment tested the IMU and algorithm with a robotic manipulator. The third experiment explores the capability of the IMU in tracking human body motion by free-hand motion.

7.1 Synthetic Data Simulation

The first experiment verified the algorithm design. Synthetic signal is generated by the following model to simulate the output from an IMU system. The model simulates a single axis rotation of one of the IMU axis at constant speed.

$$G = w_i + \eta_{g,i}, \qquad \eta_{g,i} \sim \prod(\mu_{g,i}, \sigma_{g,i}) \tag{240}$$

$$A = a_i + \eta_{a,i}, \qquad \eta_{a,i} \sim \prod(\mu_{a,i}, \sigma_{a,i}) \tag{241}$$

$$M = [\sin(\theta) \quad \cos(\theta) \quad 1] + \eta_{b,i}, \qquad \eta_{b,i} \sim \prod(\mu_{b,i}, \sigma_{b,i}) \tag{242}$$

$$i = x, y, z$$

where η are signal noise which is assumed to be Gaussian with mean (μ) and variance (σ). In addition, the signal noise model on each axis is independent from one other. The simulated raw signal is shown in Figure 7-1. The data is processed by various settings of the PF and compare to the benchmark algorithms (EKF, CF), which is summarized as shown Table 7-1.

Table 7-1 – Testing Algorithms for the experiment

Testing Algorithms													
EKF	CF	PF											
		N = 128, 256, 512, 1024											
		vMF			vMF-BC			NU			NU-BC		
		DET	RES	AUX	DET	RES	AUX	DET	RES	AUX	DET	RES	AUX

EKF: Extended Kalman Filter, CF: Complementary Filter, PF: Particle Filter, vMF: von-Mises Fisher density, NU: Non Uniform density, BC: Bias Correction, DET: Deterministic Resampling, RES: Residual Resampling,, AUX: Auxiliary Resampling

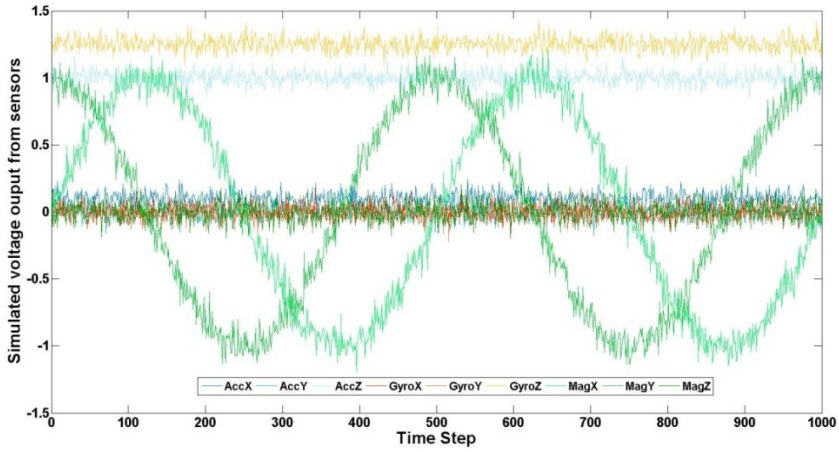

Figure 7-1 - Synthetic signal simulated raw output of the sensors

The processed data are converted back to Euler angles and compare to the ground truth, which is the intended motion trajectory without noise. The error between the algorithms' outputs and the ground truth is calculated. Figure 7-2 shows the error plot of different algorithm outputs around the axis of rotation (Z). The PFs were set up with 512 particles and auxiliary resampling was used. RAW data is the prediction based on the gyroscopes output and the observation quaternion calculated from the accelerometers and magnetometers. No feedback mechanism was used. The root mean squared errors (RMSE) were computed for each of the algorithm as shown in Figure 7-3 and Table 7-2. For the PF, the RMSEs were averaged over 200 simulation runs. The results show that increasing particle population has a positive impact to the PFs' results regardless of the resampling strategy or whether BC was used. Benchmark testing algorithms, CF and EKF, have the RMSEs of 0.93^{o}, 0.59^{o}, 0.72^{o} and 1.34^{o}, 0.71^{o}, and 0.72^{o} on each axis respectively. Among all of the testing algorithms, PF using the combination of NU density, BC, AUX along with 1024 particles gives the best result. The RMSEs are 0.49^{o}, 0.44^{o}, and 0.52^{o} on X, Y, and Z axis respectively. In addition, only PFs using the AUX and BC combination with 512 particles or more can achieve better result than the benchmark algorithms.

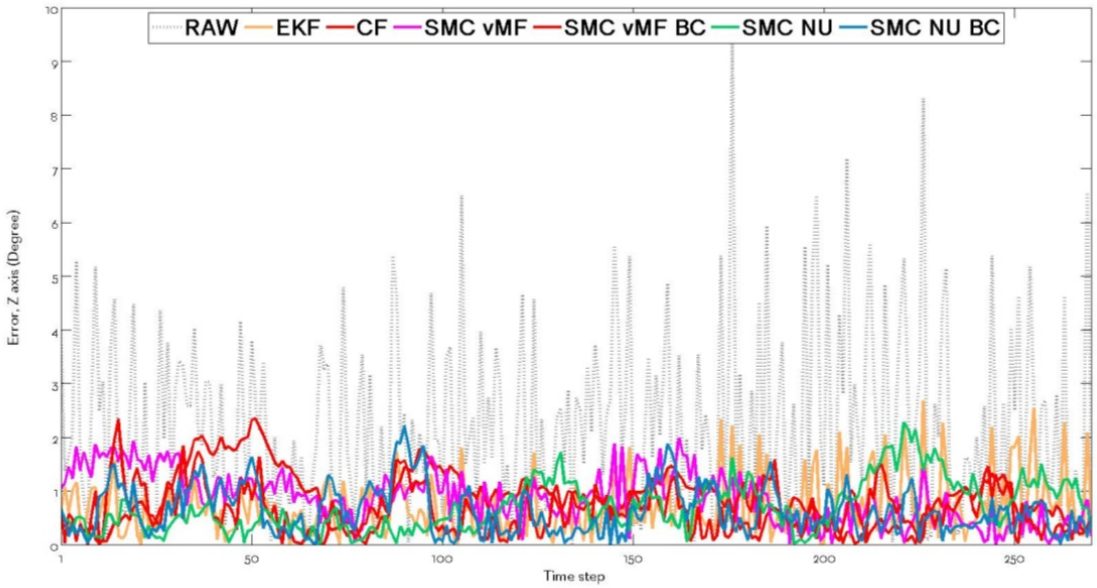

Figure 7-2 – Error between different algorithms and the intended motion

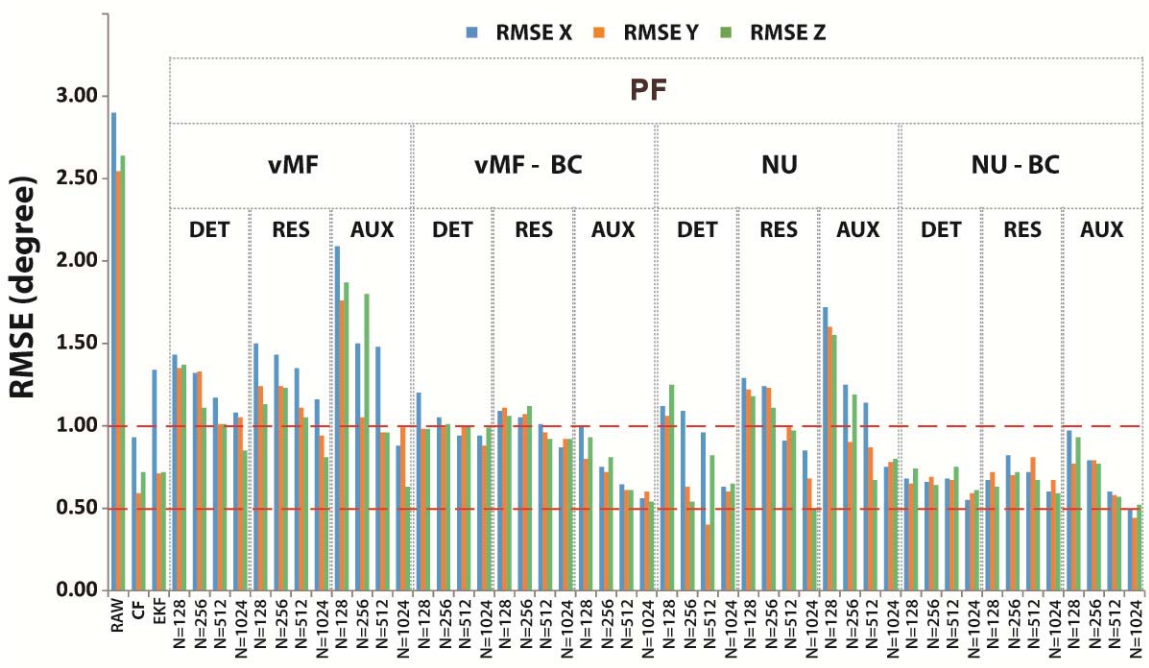

Figure 7-3 – RMSE of the testing algorithms for synthetic signal

Table 7-2 - RMSE of the testing algorithms for synthetic signal

Algorithm				RMSE X	RMSE Y	RMSE Z
Raw				2.90	2.54	2.64
CF				0.93	0.59	0.72
EKF				1.34	0.71	0.72
	Posterior Density	Importance Resampling	Particle size			
PF	vMF	DET	128	1.43	1.35	1.37
			256	1.32	1.33	1.11
			512	1.17	1.01	1.01
			1024	1.08	1.05	0.85
		RES	128	1.50	1.24	1.13
			256	1.43	1.24	1.23
			512	1.35	1.11	1.05
			1024	1.16	0.94	0.81
		AUX	128	2.09	1.76	1.87
			256	1.50	1.05	1.80
			512	1.48	0.96	0.96
			1024	0.88	0.99	0.63
	vMF BC	DET	128	1.20	0.98	0.98
			256	1.05	0.99	1.01
			512	0.94	1.00	1.00
			1024	0.94	0.88	0.99
		RES	128	1.09	1.11	1.06
			256	1.05	1.07	1.12
			512	1.01	0.96	0.92
			1024	0.87	0.92	0.92
		AUX	128	1.00	0.80	0.93
			256	0.75	0.72	0.81
			512	0.65	0.61	0.61
			1024	0.56	0.60	0.54
	NU	DET	128	1.12	1.06	1.25
			256	1.09	0.63	0.54
			512	0.96	0.40	0.82
			1024	0.63	0.60	0.65
		RES	128	1.29	1.22	1.18
			256	1.24	1.23	1.11
			512	0.91	0.99	0.97
			1024	0.85	0.68	0.50
		AUX	128	1.72	1.60	1.55
			256	1.25	0.90	1.19
			512	1.14	0.87	0.67
			1024	0.75	0.78	0.80
	NU BC	DET	128	0.68	0.65	0.74
			256	0.66	0.69	0.64
			512	0.68	0.67	0.75
			1024	0.55	0.59	0.61
		RES	128	0.67	0.72	0.63
			256	0.82	0.70	0.72
			512	0.72	0.81	0.67
			1024	0.60	0.67	0.59
		AUX	128	0.97	0.77	0.93
			256	0.79	0.79	0.77
			512	0.60	0.58	0.57
			1024	0.49	0.44	0.52

The experiment was implemented within MATLAB (Mathworks, MA) environment and was simulated with an Intel i7core mobile platform. The average processing time of each of these algorithms were shown in Figure 7-4. Both of the benchmark algorithms are extremely efficient with average of 0.0004 and 0.001 seconds per recursion. PFs took considerably longer. In addition, the processing time scales with the particle populations. PFs using vMF density require less processing time than the PFs using NU. The red dashed line in Figure 7-4 indicates threshold for reaching 30 frames per second (fps). None of the PFs with particle population over 512 require substantially longer processing time such that 30 fps refresh rate cannot be obtained.

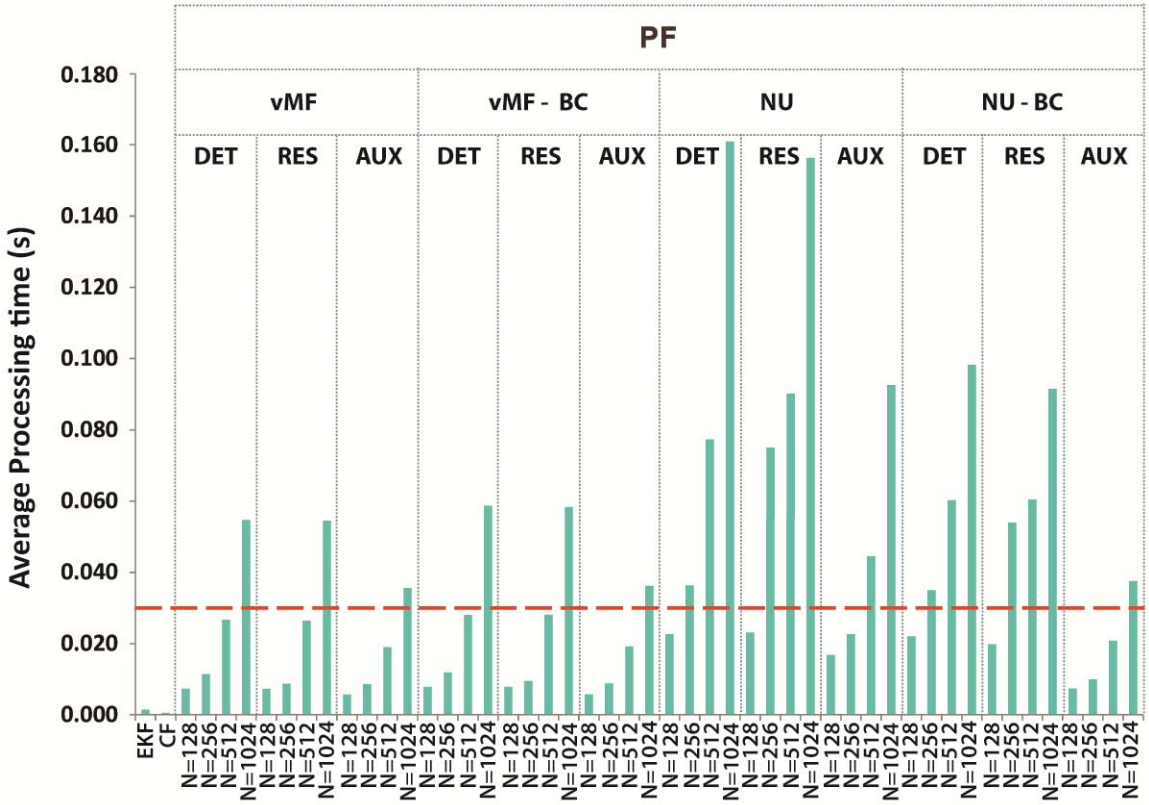

Figure 7-4 – Average processing time of the testing algorithms

Moreover, PFs using NU density with BC also consumes less processing time than the ones without. This is primarily due to the sampling method for quaternion with the NU density. Rejection sampling relies on the random quaternion proposal and produces samples that match the predetermined criteria. PF with BC dramatically reduce the search space for the optimal posterior density and converges quickly with importance sampling. PF without BC often yield suboptimal density that takes substantially longer to sample from. In Figure 7-5, the distribution on the left is the distribution obtained with PF with BC. The red, green and blue particles represent the 3-sphere projection of the accepted quaternion, and the magenta particles represent the 3-sphere projection of the quaternion rejected by the algorithm. On the right hand side, the distribution is obtained with PF without BC. As shown in the figure, the rejection algorithm rejected considerable amount of proposed quaternions, which can significantly slow down the algorithm.

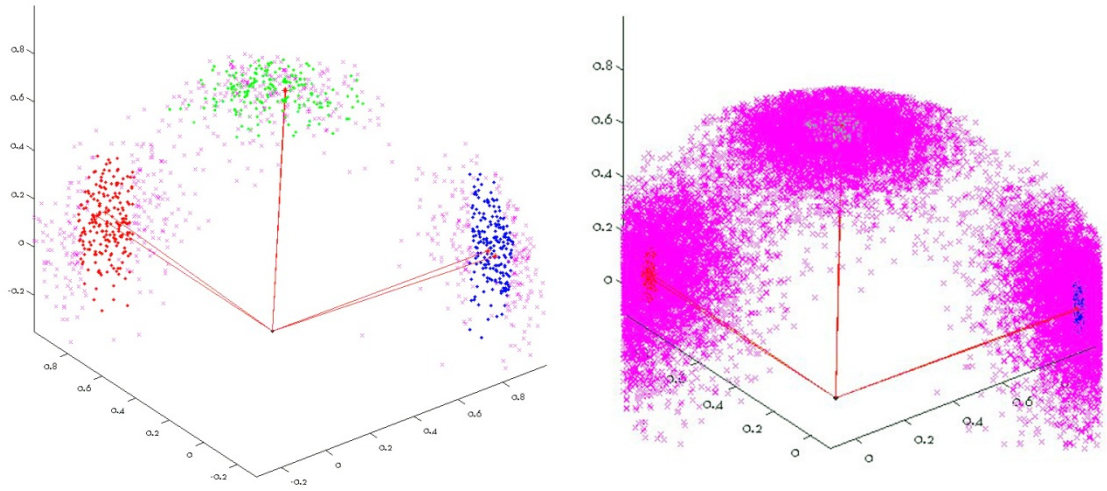

Figure 7-5 - Difference in Rejection Rate for PF with (Left) and without (Right) BC. Magenta indicates rejected particles

7.2 Robotic application

The second experiment was designed to test the hardware and software systems in a controlled setting with robotic manipulator. The modular IMU system was used and the data is processed by the testing algorithm in Table 7-1. The modular IMU was attached to a hydraulic robotic manipulator (TITAN II, Schilling robotics), as shown in Figure 7-6.

An optical tracking system is used as a reference tracking device for comparison (Polaris Spectra, Northern Digital Inc.). One of the optical trackers is attached on the robotic manipulator with the IMU and a secondary reference tracker transforms the reference co-ordinate from the camera to itself.

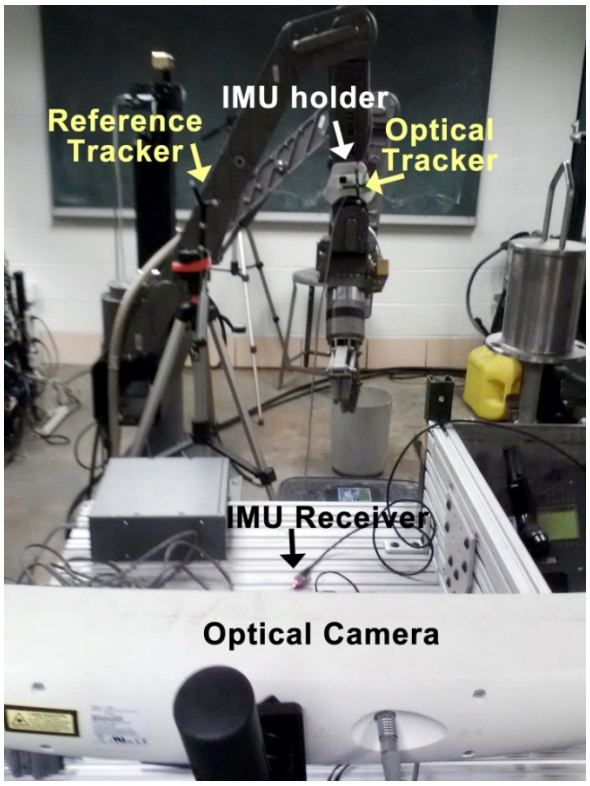

Figure 7-6 – Experimental setup with robotic manipulator

The experiments were divided into two parts. The first part of the experiment uses the manipulator to rotate the IMU into six different locations and static measurements were taken. Static measurements are defined as the unit being stationary for at least 300 samples. The reference datasets consists of 4000 sample points was collected from a stationary pose. The data were used to establish the frame of reference between the optical and IMU systems.

One of the major issues surfaced during this study is that when the robotic manipulator was powered up, a large magnetic flux was created, which caused a significantly interference to the magnetometer as shown in Figure 7-7. The generated magnetic field placed the bias very close to the limit of the sensor's dynamic range.

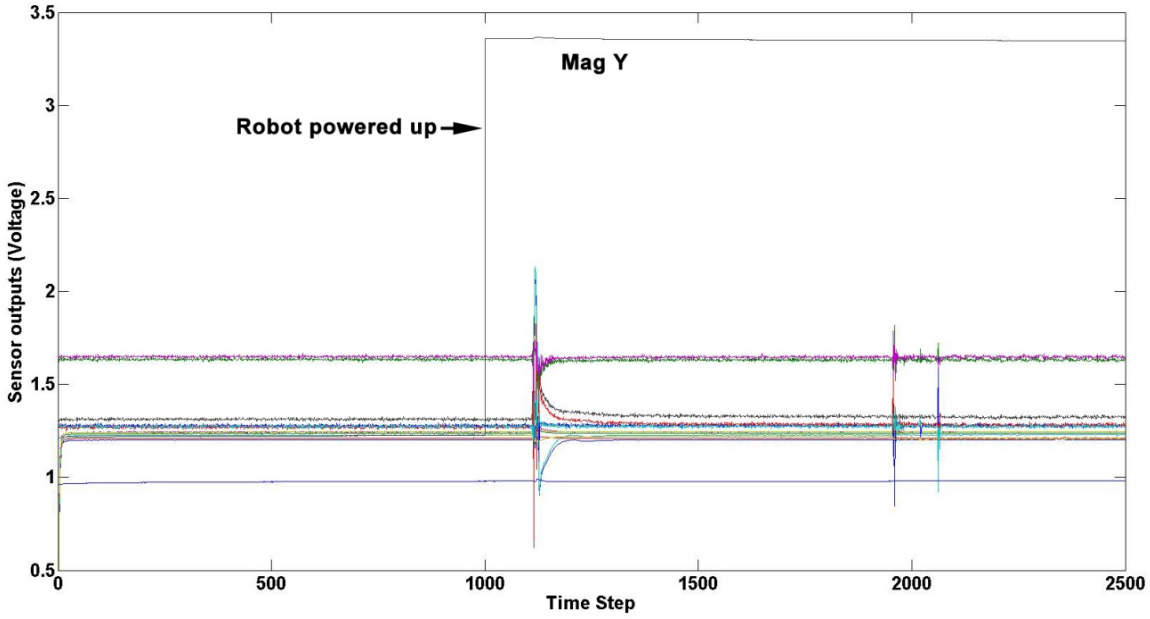

Figure 7-7 – Raw sensor output showing the magnetic interference of the robot to the magnetometer

The magnetic interference has a negative impact on the algorithm as the magnetic sensor is one of the major components to the observation quaternion calculation. In the robotic experiment, the outputs of the magnetometer were removed from the attitude estimation algorithms. This is achieved by initializing the initial azimuth direction at[0 0 1], and the azimuth rotation is only evolved by the data from the gyroscopes. Since there isn't any feedback mechanism on the azimuth, this method is subjected to drift if it's being used over an extended period of time. However, since the duration of this experiment is relatively short, it does not generate any adverse effect on the estimations.

For this experiment, there were total of six static poses with 2000 samples in each these locations. The reference and acquired data were processed by the testing algorithms. The orientation outputs in each algorithm were referenced to the quaternion calculated by the same algorithm from the reference data set. The data are then projected as Euler angles for analysis. The RMSEs between the optical tracker and the output of the IMU from the testing algorithms are shown in Figure 7-8 and Figure 7-9.

The results of the PFs were averaged over 100 simulation runs. The RMSE of all the PFs decreases as the particle population increases. Unlike the previous simulation study, the AUX importance resampling performs poorer in general against DET and RES resampling methods except for the setup with the combination of NU density and BC. In general, PFs using BC has a substantial reduction in RMSE compared to the PFs without BC. Moreover, PFs using NU density and BC performs the slightly better than other PFs' setup as shown in Table 7-3. In addition, all the PF in that setup yields very similar results regardless of particle size. According to these data, the optimal density is similar to the dispersion shape of a non-uniform distribution if the bias state is being monitored. However, without BC, vMF density yields slightly better results.

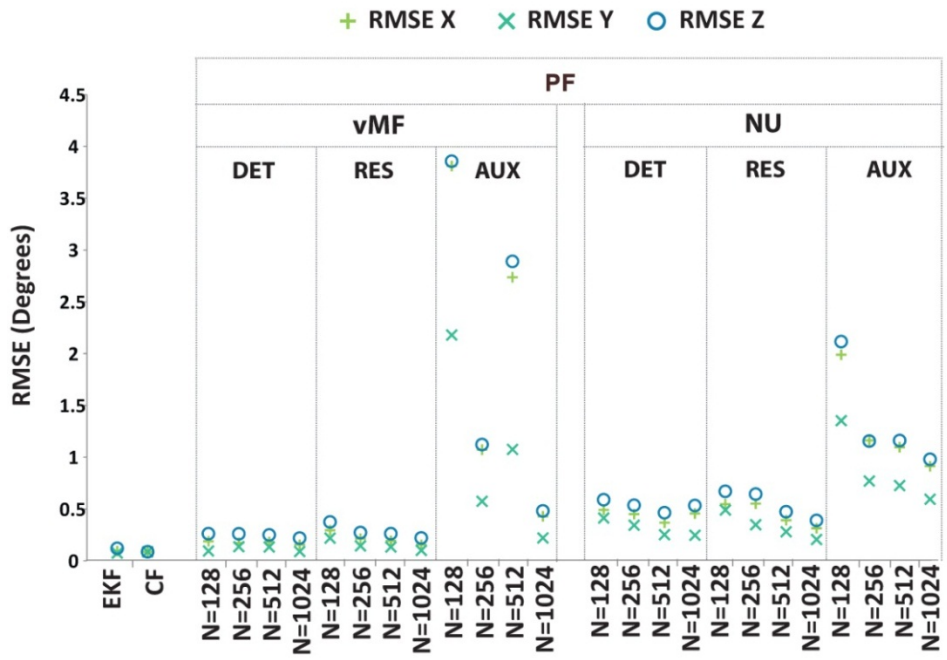

Figure 7-8 – RMSE of PF without BC against benchmark algorithms in static experiment

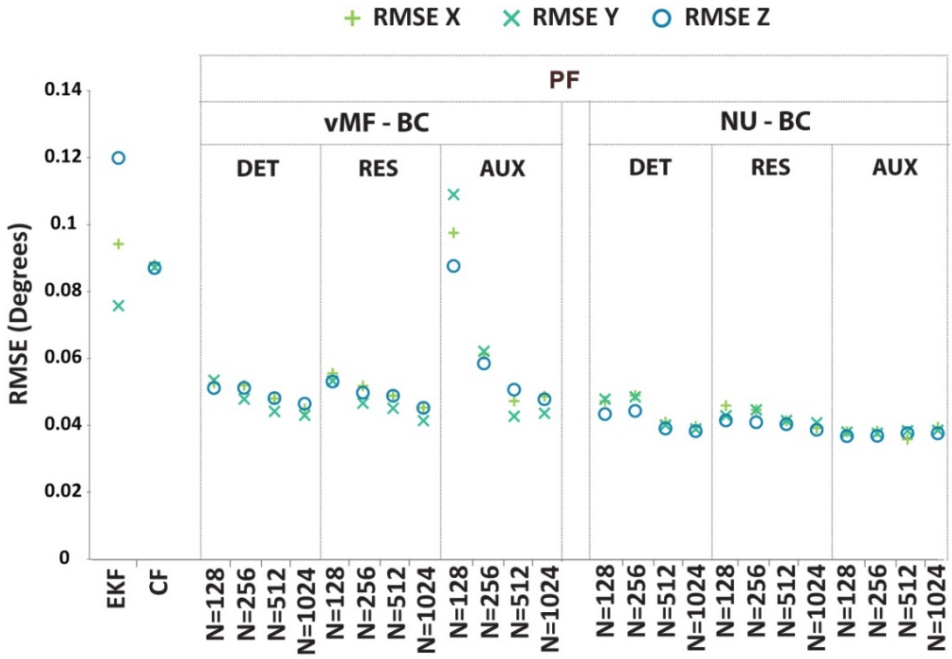

Figure 7-9 – RMSE of PF with BC against benchmark algorithms for static experiment

Table 7-3 - RMSE of the testing algorithms for static testing (Robotic)

Algorithm				RMSE X	RMSE Y	RMSE Z
CF				0.094222	0.07576	0.119904
EKF				0.08794	0.087234	0.086992
	Posterior Density	Importance Resampling	Particle size			
PF	vMF	DET	128	0.188319	0.095297	0.261448
			256	0.172609	0.13697	0.259736
			512	0.181397	0.133245	0.250197
			1024	0.15561	0.090218	0.218639
		RES	128	0.295108	0.218929	0.374347
			256	0.214258	0.144711	0.272441
			512	0.191978	0.133263	0.261605
			1024	0.155693	0.099802	0.219879
		AUX	128	3.810779	2.179338	3.857003
			256	1.07336	0.574956	1.122582
			512	2.73659	1.075031	2.889297
			1024	0.428951	0.220149	0.480787
	vMF BC	DET	128	0.052704	0.053494	0.05114
			256	0.051976	0.047886	0.051198
			512	0.048029	0.044188	0.048131
			1024	0.044953	0.043039	0.046477
		RES	128	0.055541	0.053279	0.053093
			256	0.051763	0.046638	0.049745
			512	0.048876	0.045114	0.048815
			1024	0.04526	0.041431	0.045239
		AUX	128	0.097537	0.108991	0.087638
			256	0.061223	0.062119	0.058472
			512	0.03416	0.029248	0.020629
			1024	0.048727	0.043644	0.047794
	NU	DET	128	0.492502	0.413593	0.588879
			256	0.450647	0.344851	0.535529
			512	0.367797	0.252407	0.463451
			1024	0.455673	0.247375	0.533236
		RES	128	0.546932	0.490291	0.671212
			256	0.552253	0.348043	0.643582
			512	0.39228	0.280542	0.472807
			1024	0.313518	0.205091	0.38964
		AUX	128	1.98869	1.352399	2.116004
			256	1.161698	0.770032	1.155362
			512	1.096437	0.726698	1.160275
			1024	0.913441	0.594777	0.979783
	NU BC	DET	128	0.047275	0.047833	0.043357
			256	0.048932	0.048564	0.044298
			512	0.040981	0.040113	0.03904
			1024	0.039741	0.038944	0.038255
		RES	128	0.04592	0.042901	0.041453
			256	0.044662	0.044657	0.040909
			512	0.041395	0.041463	0.040345
			1024	0.039235	0.0408	0.038686
		AUX	128	0.038039	0.037987	0.036783
			256	0.038241	0.03766	0.036878
			512	0.031382	0.039987	0.037308
			1024	0.039471	0.038537	0.037606

The second part of the robotic experiment is to verify the tracking ability of the modular IMU. The robotic manipulator was programmed to maneuver the segment where the IMU and the optical tracker were attached to. Similar to the static experiment, the relative orientations are referenced to the reference orientations calculated in previous static experiment. Figure 7-10 compares one of the 3-sphere projections of the reconstructed orientations from the PF with the optical data. In the figure below, there are several occasions that the manipulator remains stationary for an extended period. Those data should be eliminated from the analysis to reduce the influence from the static data on a dynamic experiment. This is performed by manually removing the static segment from the data after time synchronization between the optical and IMU data.

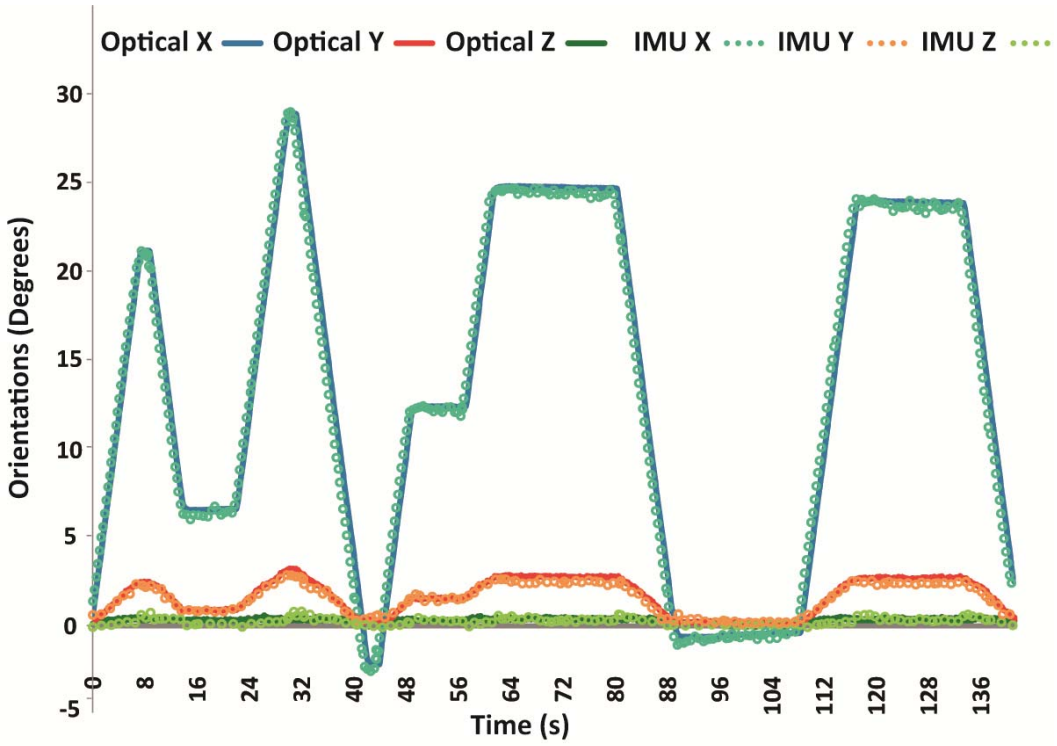

Figure 7-10 – Comparison between IMU data reconstructed with PF with the combination of NU density, BC and 256 particles population and the optical data.

The RMSE of the testing algorithms are shown in Figure 7-11 and Figure 7-12. The RMSE of the PFs are also average from 100 simulation runs. In this experiment, the RMSE of all the algorithms are slightly elevated compared to the static experiment. The PFs without BC performs relatively poorly compare to other algorithms. In addition, the PFs using AUX without BC have the highest RMSE among all testing algorithms, with most of them exceeding one degree of RMSE. On the other hand, the PFs with BC perform slightly better than the benchmark algorithms. Unlike the static experiment, the best setup combination for PF estimation is to use NU density with DET resampling and particle population of 1024 as shown in Table 7-4.

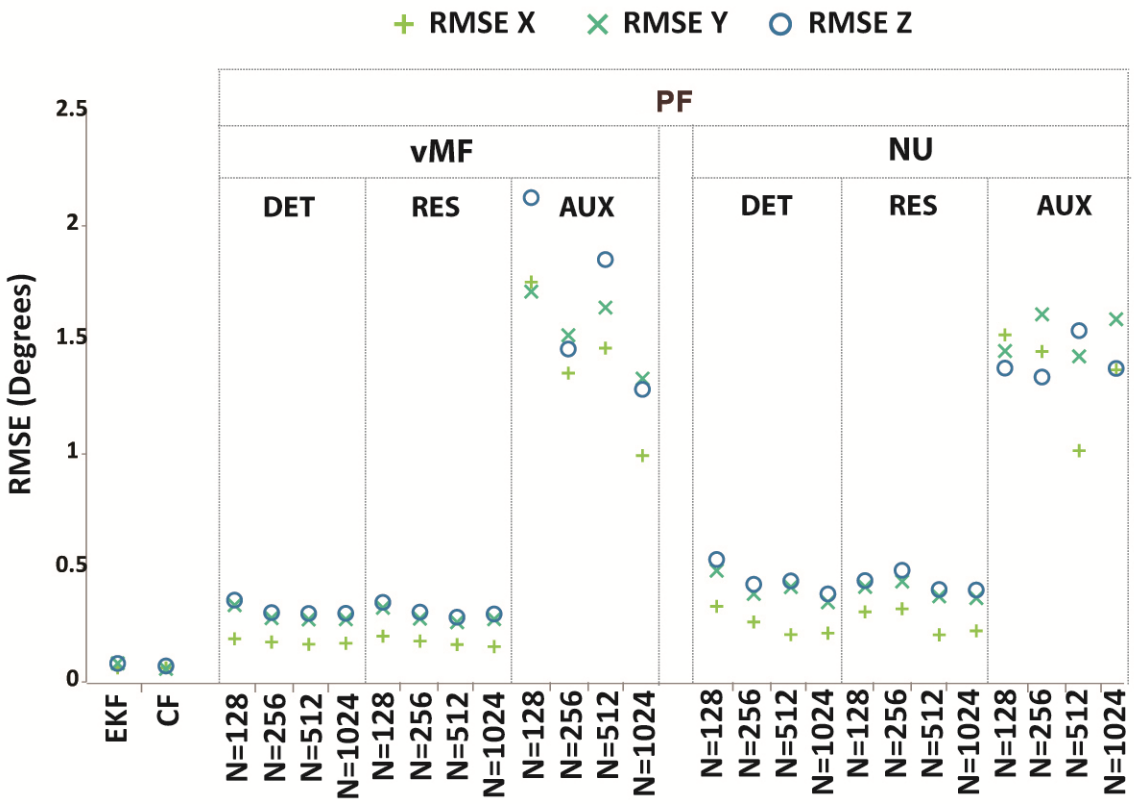

Figure 7-11 – RMSE of PF without BC against benchmark algorithms in dynamic experiment

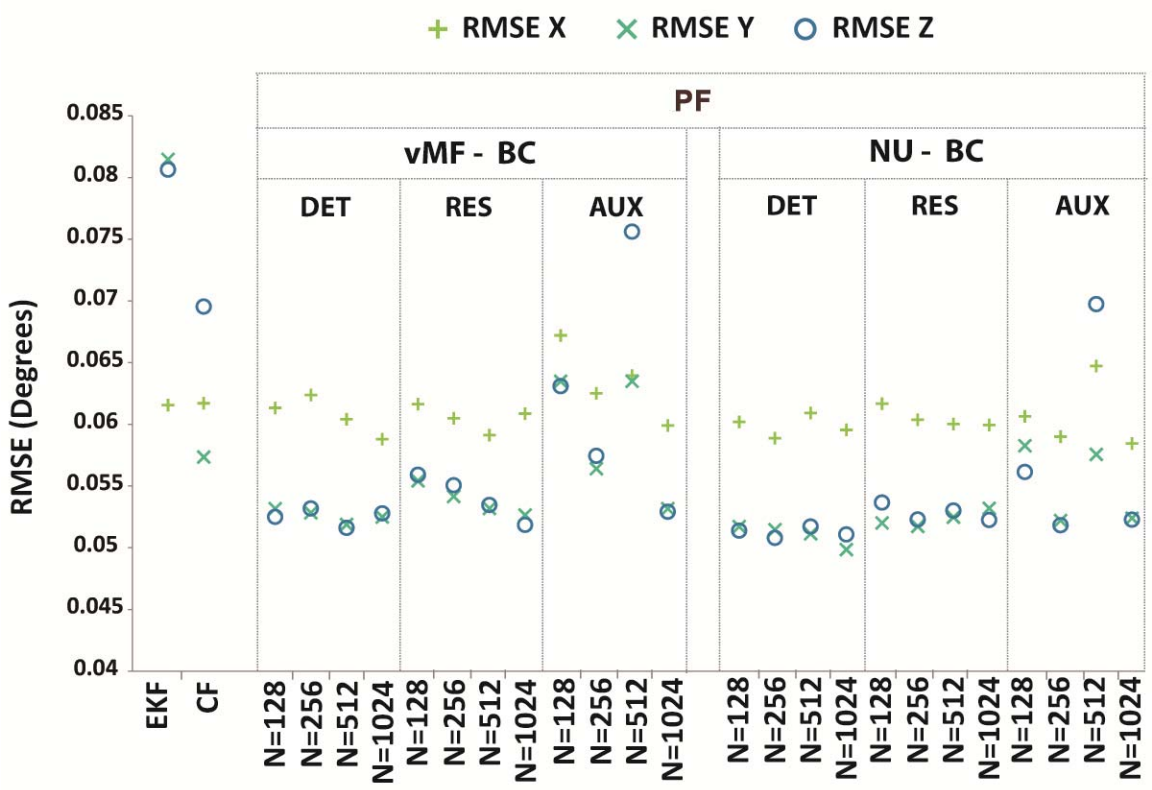

Figure 7-12 – RMSE of PF with BC against benchmark algorithms in dynamic experiment

Apart from the magnetic interference discussed in the section above, another potential source of error comes from data synchronization. Recall in Figure 6-6, the transmission time between the end devices and the access point are not constant due wireless or other interferences. Transmission lag extends $\Delta \boldsymbol{t}$ in orientation prediction equation (191), and cause the algorithm to produce wrong estimation. This is an inherent problem with using gyroscopes in the IMU since the gyroscopes' estimation is rate dependent. Although all the algorithms have feedback mechanism to correct wrong prediction, the algorithm will still need multiple recursions to converge to the optimal estimation. Additionally, the time stamps of the estimation will be offset from the optical data in an entirely random fashion. The current solution is to perform a rough data resampling based on the length of the IMU and optical dataset, and use the receiving time interval to determine if additional resynchronization is necessary in separate interval.

Table 7-4 - RMSE of the testing algorithms for dynamic testing (Robotic)

Algorithm				RMSE X	RMSE Y	RMSE Z
CF				0.0616998	0.0573525	0.0695404
EKF				0.0615615	0.0814767	0.0806343
	Posterior Density	Importance Resampling	Particle size			
PF	vMF	DET	128	0.189092	0.336857	0.358421
			256	0.174758	0.280237	0.303821
			512	0.165289	0.273752	0.299744
			1024	0.170001	0.275147	0.300706
		RES	128	0.200679	0.324983	0.348563
			256	0.17922	0.277551	0.305678
			512	0.164033	0.26233	0.284008
			1024	0.155074	0.275534	0.297549
		AUX	128	1.753507	1.711711	2.123659
			256	1.354614	1.520028	1.459685
			512	1.4640955	1.6417059	1.8519901
			1024	0.992514	1.330192	1.282388
	vMF BC	DET	128	0.061333	0.053166	0.052505
			256	0.06237	0.052839	0.053174
			512	0.060406	0.051889	0.051606
			1024	0.058796	0.052492	0.052777
		RES	128	0.061639	0.055421	0.055916
			256	0.060491	0.054152	0.055051
			512	0.059117	0.053177	0.053464
			1024	0.060864	0.05265	0.051852
		AUX	128	0.06721	0.063484	0.063084
			256	0.062515	0.056407	0.057441
			512	0.063941	0.063491	0.075611
			1024	0.059904	0.05317	0.05291
	NU	DET	128	0.332174	0.487942	0.536764
			256	0.262747	0.386814	0.427934
			512	0.207468	0.415915	0.442872
			1024	0.214491	0.349704	0.3855
		RES	128	0.307349	0.416816	0.444533
			256	0.321066	0.440656	0.489518
			512	0.20685	0.375547	0.40531
			1024	0.223702	0.368107	0.40286
		AUX	128	1.521842	1.451801	1.376179
			256	1.449068	1.612242	1.336461
			512	1.014376	1.427508	1.540379
			1024	1.368266	1.590364	1.374839
	NU BC	DET	128	0.060193	0.051715	0.051381
			256	0.058877	0.051457	0.050797
			512	0.060915	0.05113	0.051715
			1024	0.059541	0.049843	0.051075
		RES	128	0.061677	0.052007	0.053656
			256	0.060378	0.051723	0.052286
			512	0.060025	0.052474	0.053012
			1024	0.059948	0.053195	0.052247
		AUX	128	0.060641	0.058272	0.056142
			256	0.058997	0.052201	0.051822
			512	0.064721	0.057566	0.06973
			1024	0.058446	0.052396	0.052275

7.3 Human motion tracking

The last validation experiment was to use the modular IMU and test the algorithms with human body motion. This is achieved by performing orientation tracking of the IMU with free hand motion activities. The estimated orientation from the IMU is compared to the optical system. A plastic container was created with rapid prototyping machine to fit the IMU and passive optical marker (Figure 7-13). In addition, the container has additional mounting spots for more optical trackers to provide flexibility in testing the system such that it will not restricted by the line of sight limitation of the optical tracker.

In this validation study, 35 free hand motion activities were performed. Similar to the robotic experiment, this data acquisition is divided into two parts. The first part of the activity collects static reference data, and the second part of the activity collects dynamic motion data. The protocol on the following page was used for this activities testing.

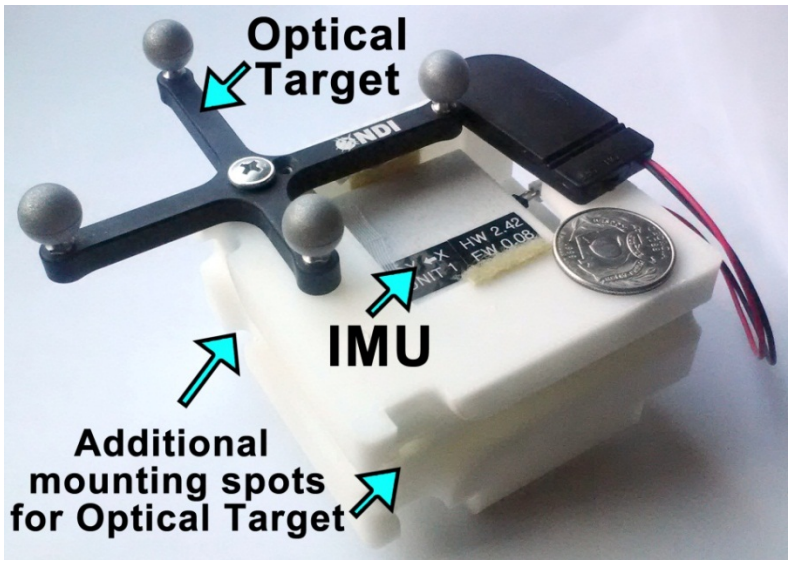

Figure 7-13 – Plastic container for both the IMU and optical systems for free hand motion

1. Holding the container such that one of the optical trackers is facing the optical camera sensors and maintain the pose as steady as possible for 4000 samples. This is used as the human static testing as well as creating the reference pose for the activity.
2. The user is free to maneuver the container after step 1 (Figure 7-14)
3. In order to facilitate the data synchronization procedure discussed in the robotic experiment, the subject is encouraged to pause the motion briefly every 1500 samples. The data acquisition progress and the 1500-sample mark are shown on the computer. This step is not necessary, but it can help with the resynchronization of out of sync data during analysis.
4. At the end of each acquisition, the subject holds the container still for a brief period of time (~200 samples) before exiting the acquisition of the IMU and optical tracking system.

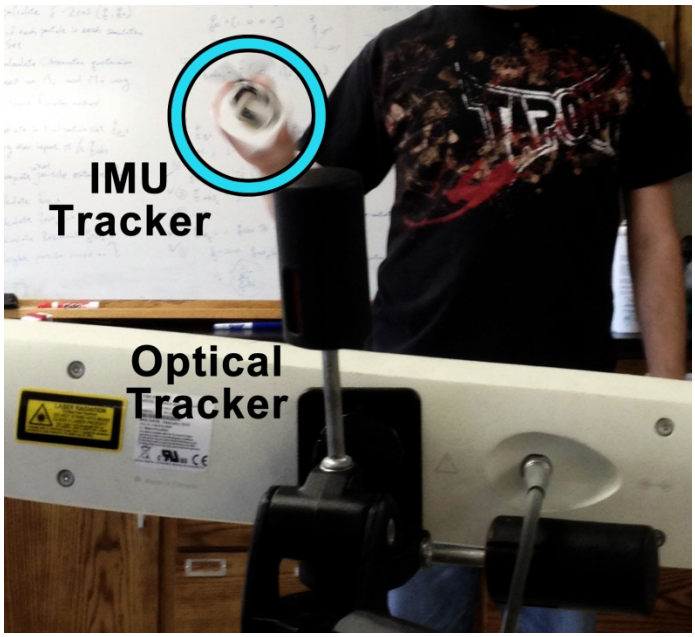

Figure 7-14 – Free hand motion testing

The optical data were checked to see if there is any corrupted data caused by interference or the tracker is partially out of view from the optical camera. If amount of corrupted data is small, interpolated data is used as replacement. However, if a large portion of the data is corrupted, the section is noted to remove from analysis.

One of the interesting discoveries during the static experiment was the signal characteristic changed considerably compare to the previous experiments. It was expected that the variances of the signals is higher for human body motion due to the feedback mechanism of the musculoskeletal system. However, it was not expected that the elevation in signal variances are not uniform among the sensors. As shown in Figure 7-15, the change in variance for the accelerometers was substantially higher than the gyroscopes and magnetometers. While it should not present any problem to the Kalman class estimation family, it becomes problematic for the complementary filter class. This is due to the assumptions of the CF discussed in Figure 2-3. The core idea of CF is assuming the gyroscopes data has a higher frequency noise than the accelerometers and magnetometers, and the two datasets complement each other by the CF. While this assumption is still valid for robotic tracking application as shown in Figure 7-15, it does not fit with the data obtained in human motion analysis. This is very likely the fundamental cause of the instability issues observed with the CF algorithm.

In the free hand motion study, instability has been observed on both of the benchmark algorithms. There were 5 instances of instability observed with CF algorithm and 2 partial cases of instability observed with the EKF algorithm. There are zero cases of instability from any of the setups of the PFs. All of the instability cases with CF failed to stabilize during the initialization and static phases of the experiment, and the filter never converges throughout the entire activity. Figure 7-16 shows one of the activities where CF did not stabilize and Figure 7-17 shows the same activity processed by PF.

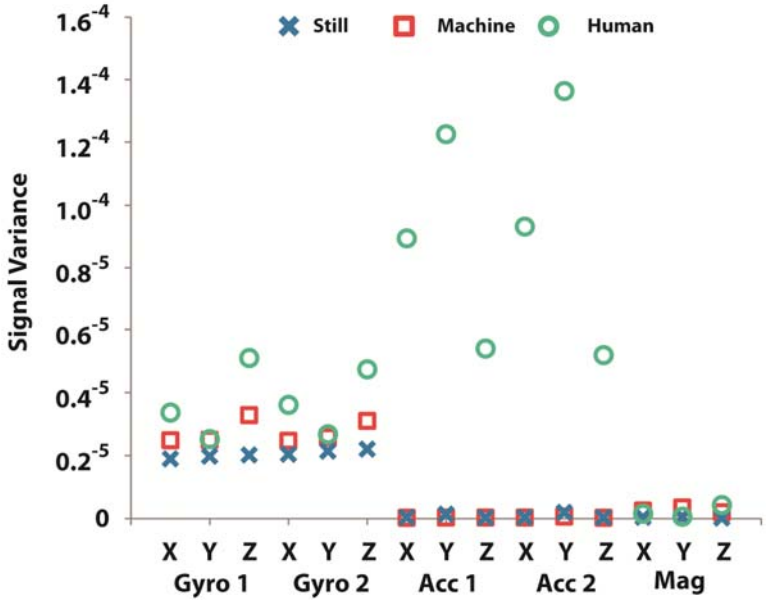

Figure 7-15 – Signal variance of the IMU tracker during still, static (robot), static (human)

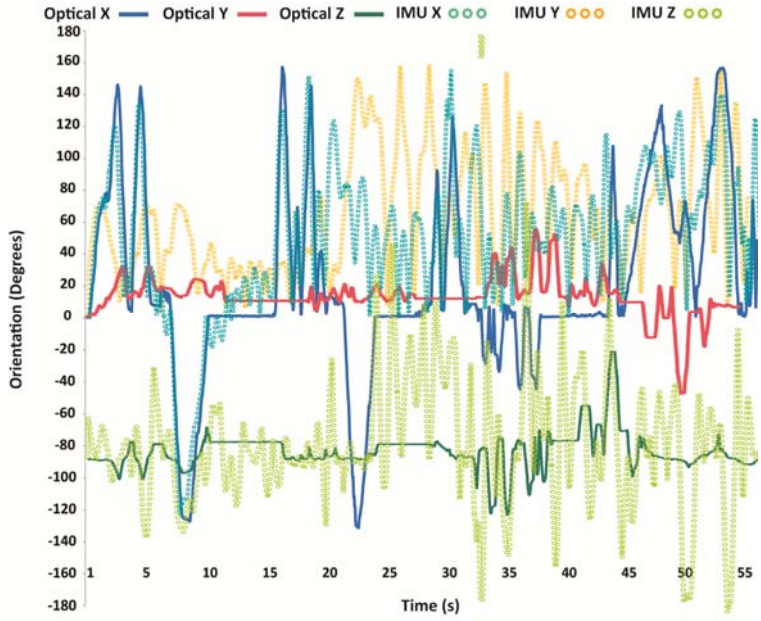

Figure 7-16 – One of the activities showing the instability of CF during human motion testing

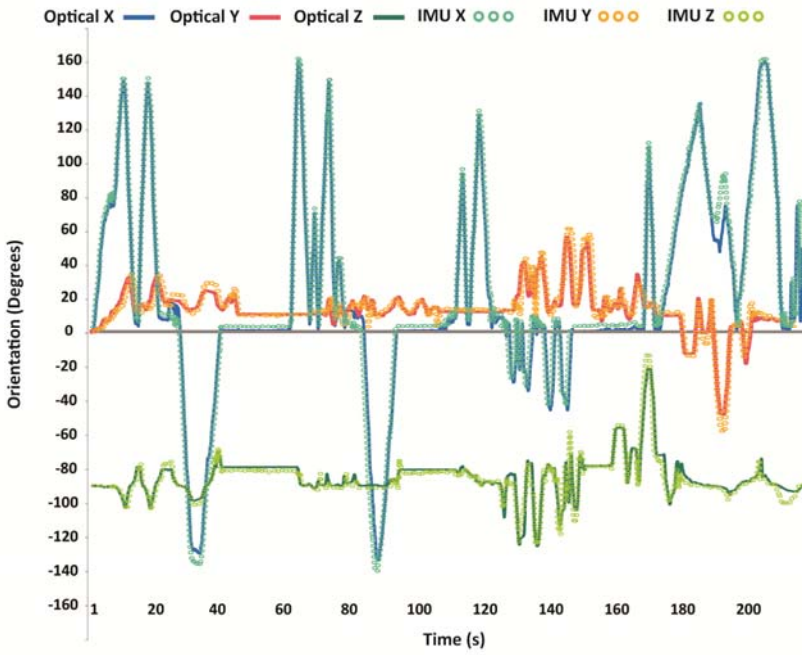

Figure 7-17 – Activity in Figure 7-16 processed with PF (NU density, BC, RES, 256 particles)

As for the instability cases of the EKF, one of the cases destabilized briefly during the dynamic activities but quickly recovers during one of the recommended time resynchronization stall from step 3 of the protocol. However, for the second case, the filter failed to stabilize throughout the entire activity as shown in Figure 7-18. The initial hypothesis for the cause of instability is the initial setup parameters for the EKF. Upon further investigation, it was discovered that there was a narrow range of the components in the process noise matrix that stabilizes the filter in this specific activity. However, the new values are significantly higher than the data that was determined empirically from previous experiments. In contrast, all the initial parameters for the PFs were setup identically in all of the tests presented. Figure 7-19 shows the output of the PF processing the same activity. In addition, the error analysis of this study omits all destabilized cases from the calculation.

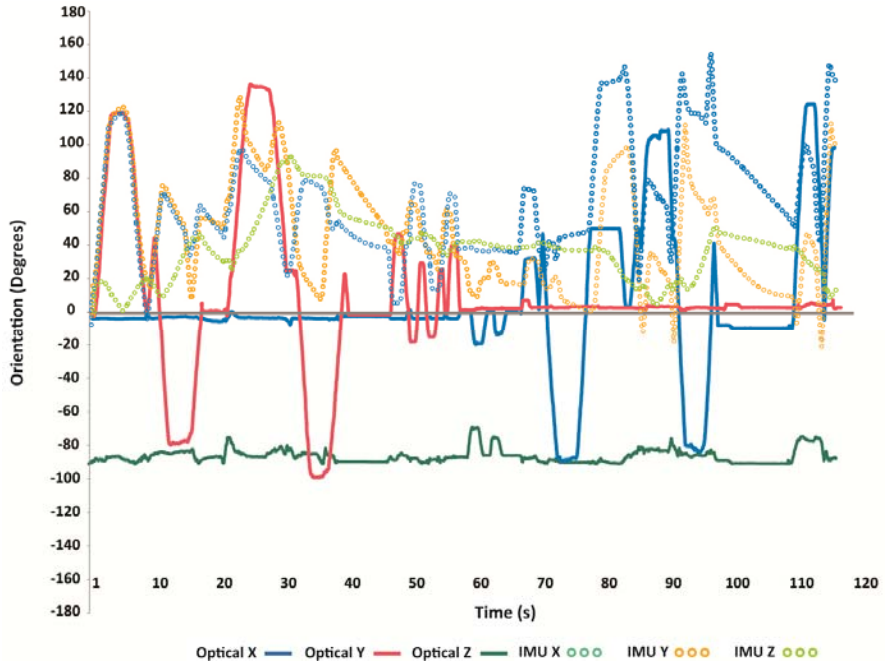

Figure 7-18 – Activity showing the instability of EKF during human motion testing

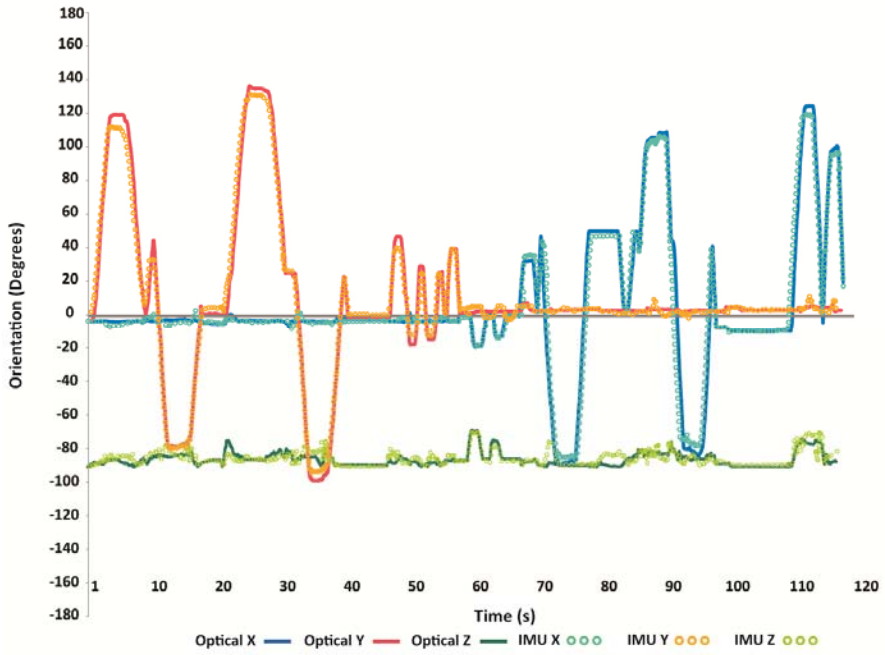

Figure 7-19 – Activity in Figure 7-18 processed with PF (NU density, BC, RES, 256 particles)

Using the data from the optical tracking system as the base reference, the RMSE of the IMU system in the static and dynamic experiment were determined. Figure 7-20 and Figure 7-21 show the RMSE of the IMU during the static experiment. As seen from the figures, the RMSE computed from all of the testing algorithms for free hand motion have significantly increased compared to the robotic experiment. Similar to the result in previous experiment, PF without BC generally performs less accurate than the ones using BC. In addition, the performance of PF using AUX resampling without BC is the least accurate among all testing algorithms. On the other hand, all other PFs' setups perform as well or better than the EKF benchmark algorithm. However, only PFs using the combination of BC and DET/RES resampling can perform as good as the CF.

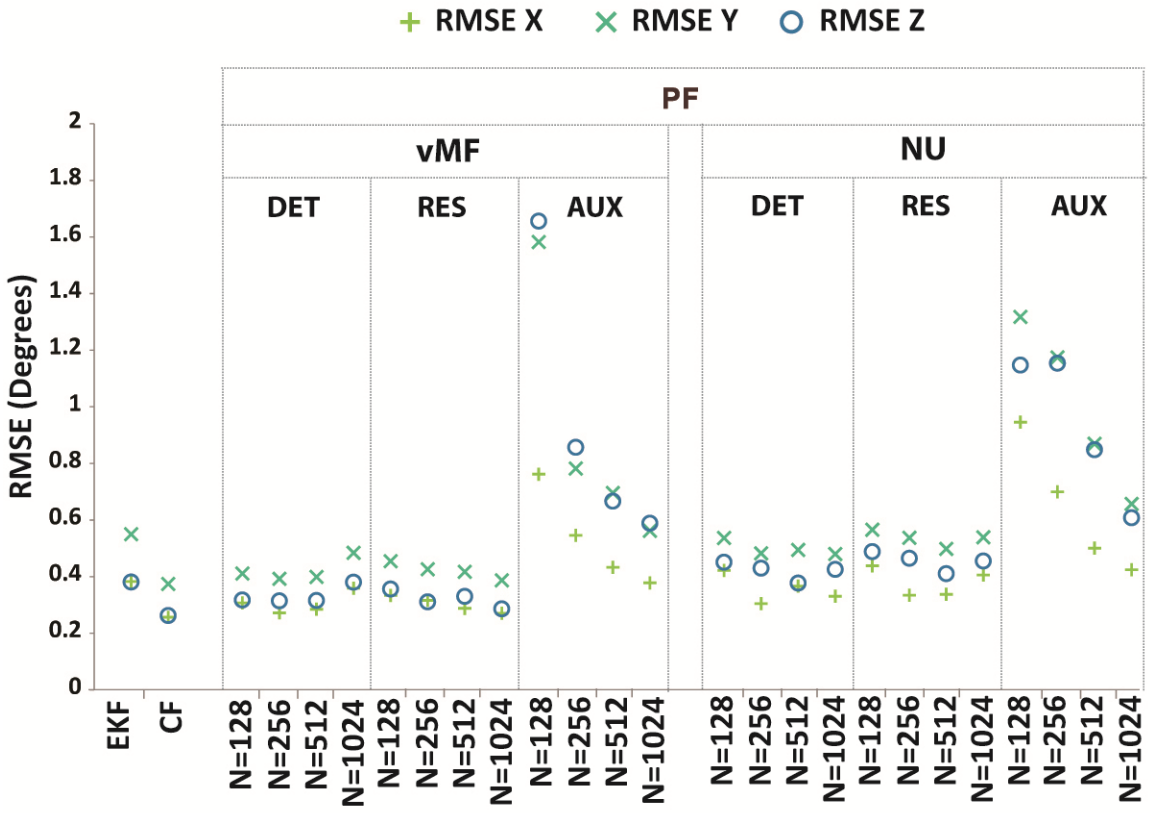

Figure 7-20 – RMSE of PF without BC against benchmark algorithms in static experiment

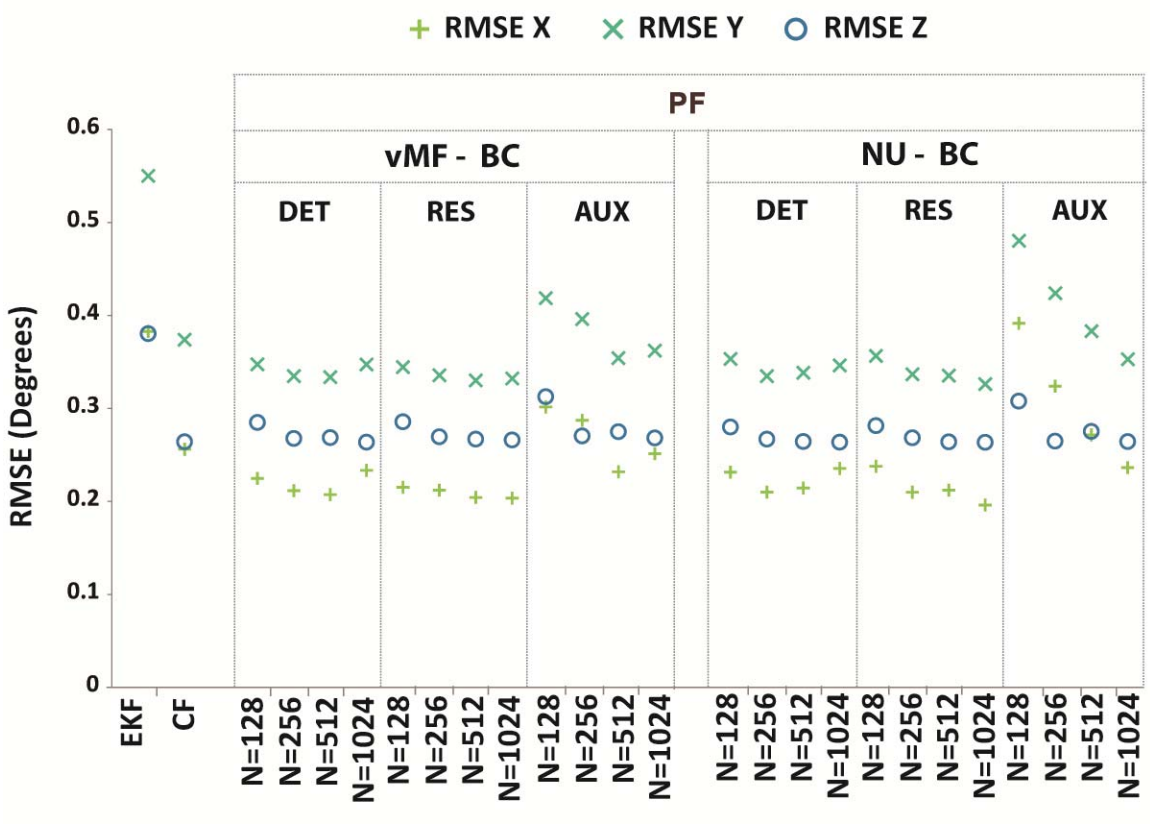

Figure 7-21 – RMSE of PF with BC against benchmark algorithms in static experiment

As shown in the figures, the size of the particle population still affects the accuracy of the estimation considerably for PFs using AUX resampling. Significant improvement can be seen as the particle population increases. However, the effect of the particle size is more subtle in other PF setups. The numerical values of the RMSE of all the tested algorithms are shown in Table 7-5. Among all the testing algorithms, PF using the combination of NU density, BC, and RES resampling method along with particle size of 1024 particles achieved the best estimation result relative to the optical tracking system.

Table 7-5 - RMSE of the testing algorithms for static testing (Free hand)

Algorithm				RMSE X	RMSE Y	RMSE Z
CF				0.25612	0.373993	0.262638
EKF				0.382841	0.550061	0.380544
	Posterior Density	Importance Resampling	Particle size			
PF	vMF	DET	128	0.307122	0.410819	0.317397
			256	0.271844	0.392004	0.314143
			512	0.284503	0.398816	0.315527
			1024	0.358976	0.484019	0.380054
		RES	128	0.333785	0.45429	0.355885
			256	0.315931	0.425939	0.310733
			512	0.287602	0.417071	0.330296
			1024	0.270091	0.387038	0.286044
		AUX	128	0.761925	1.582088	1.655977
			256	0.545583	0.781831	0.856825
			512	0.432447	0.695762	0.666667
			1024	0.37797	0.561692	0.588535
	vMF BC	DET	128	0.224862	0.34756	0.285018
			256	0.211572	0.33483	0.267825
			512	0.207415	0.333673	0.268589
			1024	0.233448	0.347217	0.263668
		RES	128	0.215172	0.34466	0.285647
			256	0.212124	0.335681	0.269624
			512	0.204367	0.330324	0.267061
			1024	0.203573	0.332325	0.266135
		AUX	128	0.301573	0.41862	0.312684
			256	0.287334	0.396205	0.270464
			512	0.231889	0.354159	0.275005
			1024	0.251373	0.362193	0.268307
	NU	DET	128	0.422045	0.536371	0.450702
			256	0.304779	0.482179	0.429707
			512	0.366009	0.494424	0.377386
			1024	0.33032	0.479976	0.425423
		RES	128	0.438439	0.565644	0.488093
			256	0.334518	0.537028	0.46502
			512	0.337148	0.497906	0.41011
			1024	0.405563	0.538982	0.455208
		AUX	128	0.945622	1.317771	1.147613
			256	0.699663	1.17447	1.154055
			512	0.500278	0.869395	0.8483
			1024	0.424287	0.656811	0.608006
	NU BC	DET	128	0.231545	0.353274	0.280067
			256	0.210024	0.334776	0.267098
			512	0.214518	0.338352	0.264464
			1024	0.235435	0.346328	0.263697
		RES	128	0.23785	0.356481	0.281458
			256	0.209942	0.336734	0.268509
			512	0.212007	0.335277	0.264162
			1024	0.196174	0.326332	0.263526
		AUX	128	0.39146	0.480272	0.307755
			256	0.32398	0.424034	0.264874
			512	0.271683	0.383183	0.27531
			1024	0.236518	0.353079	0.26451

The RMSE of the dynamic testing are shown in Figure 7-22 and Figure 7-23. The RMSE of the free hand motion activities have increased in all testing algorithms. However, the performance between each algorithm is similar to the static testing. The PFs without BC does not surpass the performance of the benchmark algorithms. In addition, the estimation of PF with AUX resampling without BC is the least accurate among all testing algorithms, where most of the RMSE are over one degree. There is no significant difference between using vMF and NU densities without BC.

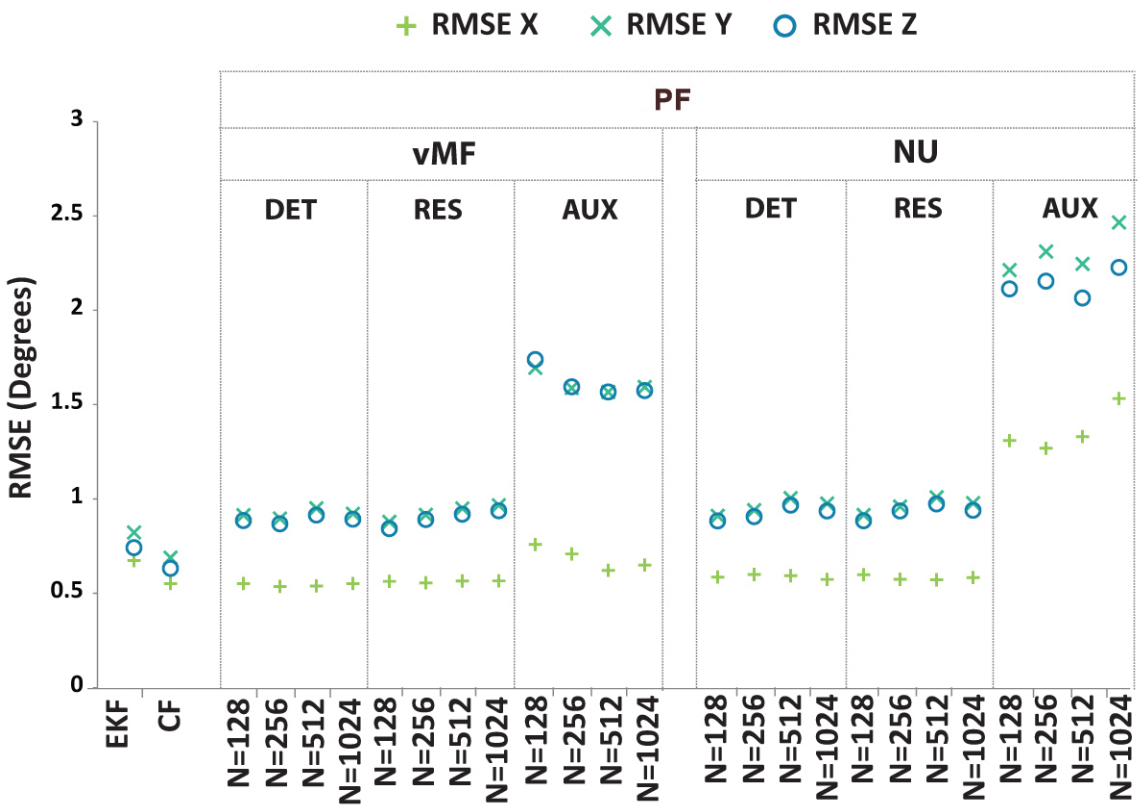

Figure 7-22 – RMSE of PF without BC against benchmark algorithms in dynamic experiment

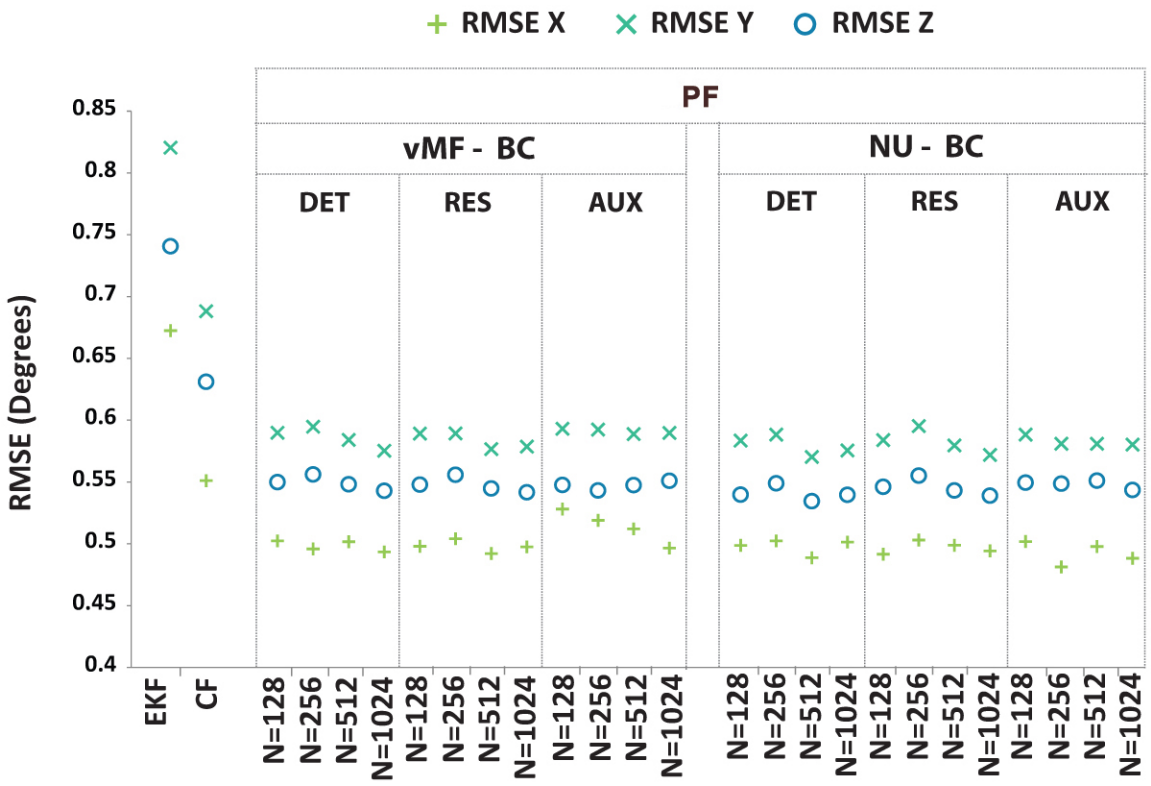

Figure 7-23 – RMSE of PF with BC against benchmark algorithms in dynamic experiment

The PFs using BC performs substantially better. All of the PFs using BC surpassed the performance of both EKF and CF benchmark algorithms. There is also no significant difference between different setups for the PFs with BC, although PF using the combination of NU density, BC with AUX resampling performs slightly better than other setups as shown in Table 7-6. Figure 7-24 shows the average processing time for one recursion for all the testing algorithms. The red dashed line indicates the limit to achieve 30 fps. The PFs takes considerably longer processing time than the benchmark algorithms. The PFs using vMF density is capable of processing the data higher than the 30fps limit up to particle size of 512.

Table 7-6 - RMSE of the testing algorithms for dynamic testing (Free hand)

Algorithm				RMSE X	RMSE Y	RMSE Z
CF				0.4952489	0.6107693	0.5665034
EKF				0.6182459	0.7526171	0.6829901
	Posterior Density	Importance Resampling	Particle size			
PF	vMF	DET	128	0.508892	0.797903	0.77175
			256	0.499949	0.785951	0.759114
			512	0.500613	0.839728	0.806064
			1024	0.499115	0.90813	0.880749
		RES	128	0.519744	0.773136	0.740493
			256	0.51388	0.800693	0.777159
			512	0.52013	0.839364	0.809415
			1024	0.52974	0.858904	0.827738
		AUX	128	0.657208	1.483329	1.508125
			256	0.616842	1.393874	1.394849
			512	0.5479825	1.3873705	1.384049
			1024	0.573341	1.412107	1.389825
	vMF BC	DET	128	0.479415	0.538749	0.505995
			256	0.475554	0.542755	0.510727
			512	0.481012	0.541729	0.506701
			1024	0.474022	0.541672	0.516435
		RES	128	0.476657	0.53817	0.504506
			256	0.481245	0.538359	0.510545
			512	0.472261	0.535864	0.504269
			1024	0.482221	0.546473	0.516422
		AUX	128	0.495023	0.542086	0.505512
			256	0.482958	0.539402	0.50105
			512	0.480376	0.537294	0.504424
			1024	0.471602	0.55182	0.522952
	NU	DET	128	0.534292	0.802687	0.776815
			256	0.545651	0.821376	0.789982
			512	0.542016	0.882812	0.84822
			1024	0.510333	0.94512	0.908605
		RES	128	0.545159	0.801632	0.774195
			256	0.527002	0.835063	0.812712
			512	0.5205	0.888143	0.855909
			1024	0.5472	0.870096	0.828992
		AUX	128	1.098755	1.903985	1.818134
			256	1.037757	1.936444	1.815091
			512	1.071673	1.885124	1.744471
			1024	1.240428	2.06301	1.876578
	NU BC	DET	128	0.475803	0.534248	0.498518
			256	0.477444	0.537339	0.505884
			512	0.465708	0.529678	0.496506
			1024	0.475375	0.540826	0.514757
		RES	128	0.468722	0.535177	0.503762
			256	0.47744	0.543751	0.510588
			512	0.474823	0.537505	0.503417
			1024	0.47921	0.539201	0.513155
		AUX	128	0.467373	0.537376	0.506471
			256	0.454624	0.529517	0.504732
			512	0.459872	0.530146	0.506857
			1024	0.457122	0.529544	0.49975

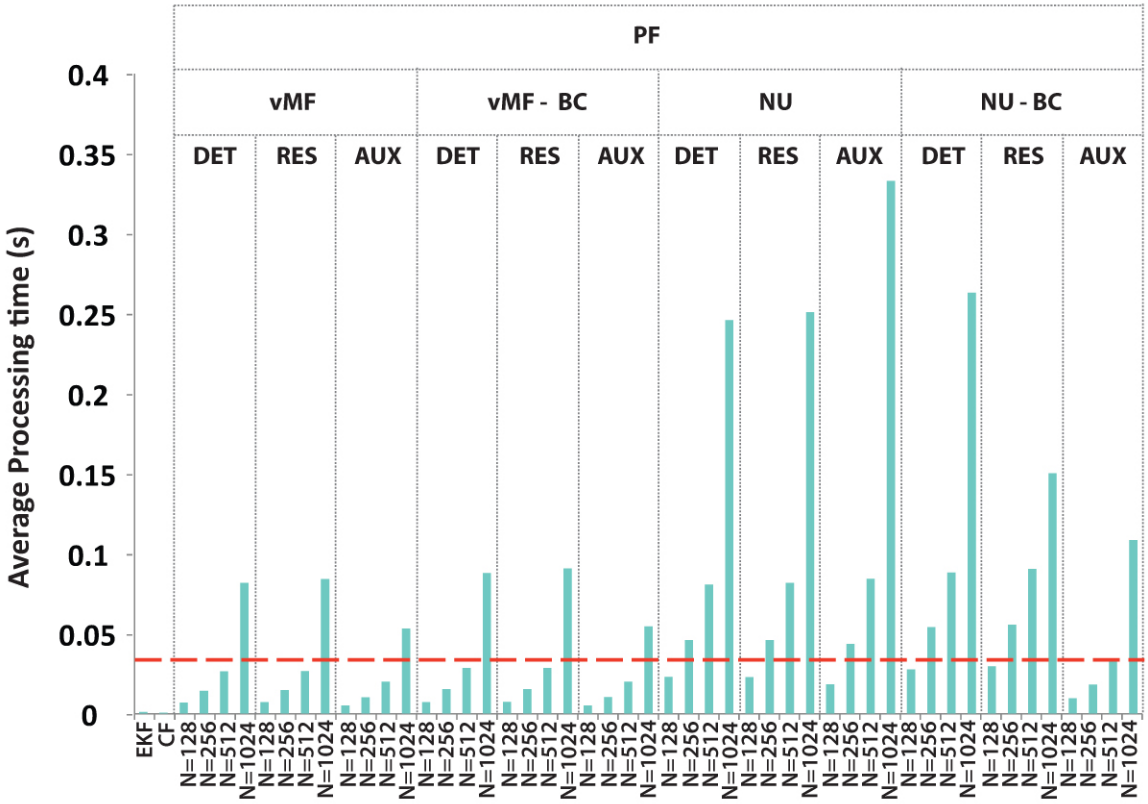

Figure 7-24 – Average processing time of the testing algorithms for free hand activities

For the PFs using NU density, only particle size of 128 is capable of achieving higher than 30 fps limit except the combination with BC and AUX resampling, where up to 512 particles can processing the data within 0.033 seconds per recursion. Based the error analysis and the processing time of the testing algorithm, the PF using the combination of NU density, BC, and AUX resampling with 256 particles is most suitable for freehand motion activities.

7.4 System analysis

Recalling the initial experiment with OTS IMU in chapter 4, the EKF algorithm that was used as benchmark algorithm for the robotic and free hand experiment in this

chapter is the same implementation of the EKF used for the OTS IMU assessment in chapter 4. The only difference was the process noise covariance was updated. Due to the change and upgrade of the hardware components, the modular IMU demonstrates significant improvement for orientation tracking accuracy summarized in Table 7-7.

There are several reasons why the results of the testing during the robotic experiment are far better than the free hand motion. The motion of the robotic manipulator is much smoother than the human free hand motion. As mentioned previously, the neuron feedback mechanism of the human musculoskeletal system alters the signal characteristic of the sensors output.

Another cause of error comes from the magnetometer. Due to the magnetic interference during the robotic experiment, the outputs from the magnetometers were not used in the attitude estimation algorithm. However, it was used in the free hand motion study. It was discovered that one of the magnetometers axis exhibits cross axis effect once the sensor is slighted tilted. Cross axis calibration were performed to minimize the effect initially by rotating the IMU around all three of the major axis. The outputs of the magnetometer were analyzed and the cross axis coefficients were determined empirically. The raw output of the X axis and Z-axis azimuth detection is slightly elliptic as shown in Figure 7-25. Cross axis correction is applied prior to offset and scale compensation. Figure 7-26 shows the corrected outputs of the magnetometers.

Table 7-7 – Comparison orientation accuracy between OTS and Modular IMU (Both are processed by EKF)

	RMSE X	RMSE Y	RMSE Z
OTS IMU (average)	3.64^{o}	4.18^{o}	3.58^{o}
Modular IMU (average)	0.62^{o}	0.75^{o}	0.68^{o}

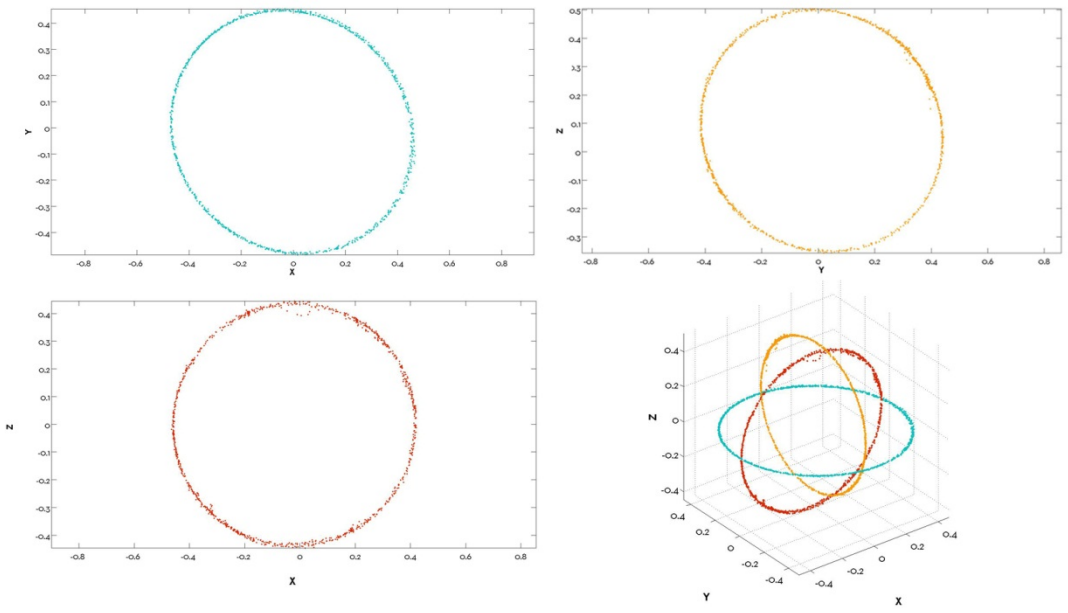

Figure 7-25 – Raw magnetometers output (Blue: Z axis, Red: Y axis, Orange: X axis)

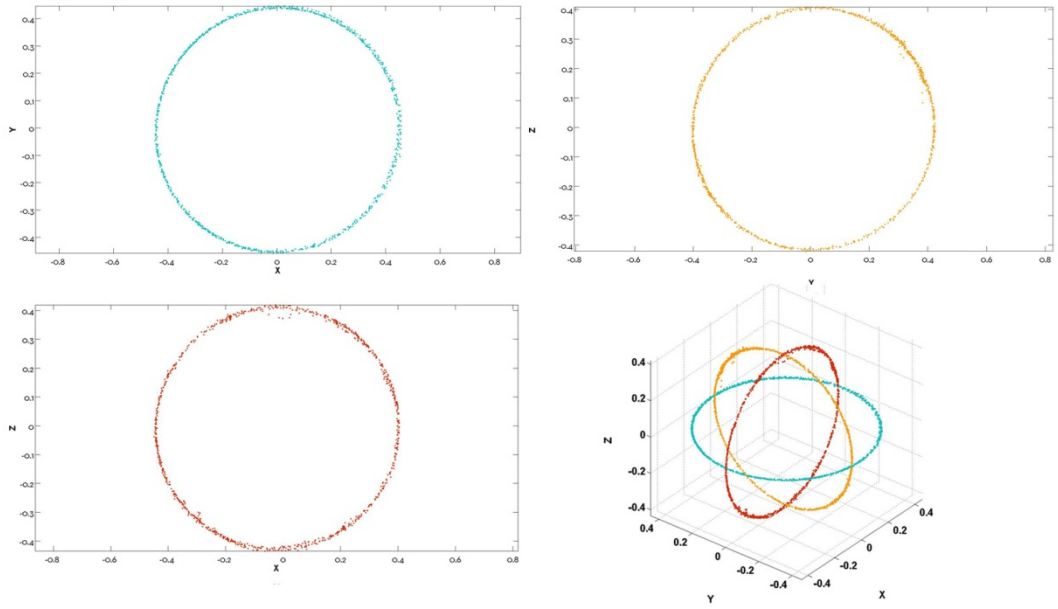

Figure 7-26 – Corrected magnetometers output (Blue: Z axis, Red: Y axis, Orange: X axis)

Despite of the calibration, a secondary artifact was observed on the rotation perpendicular to the magnetic field around Y-axis observed at certain azimuth direction. During this motion, the magnetometers use the orientation estimation from previous state to compensate the outputs with the declination process. However, due to this artifact, the outputs of magnetometers do not declinate correctly and cause error in the estimation. Since there is no effective method to monitor and compensate this error without extensive calibration with a multi-axis rate table at each testing environment, the current approach is to strategically place the IMU such that Y-axis is parallel to the axis with least motion. In addition, the magnetometer feedback is disabled if the azimuth is point in the direction near the artifact region.

8. Biomedical Applications

The primary function for IMU orientation tracking in biomedical applications is to assist diagnosis. Since many diseases are diagnosed by observing the joint mechanics of the patients, IMU device can be used to provide quantitative results for the physicians and medical device engineers. The following studies demonstrate the capability of the modular IMU as a diagnostic device. The first study is similar to the experiment conducted in the pilot study in Chapter 4.6. The goal is to monitor the joint kinematics during flexion and extension activities such as deep knee bend and chair rise. The second study demonstrates the potential for IMU as a diagnostic device for lumber spines' condition.

8.1 Knee joint activities

The primary focus in the knee joint research is to observe the kinematics of the joint during motion and to apply the findings in designing the treatment plans or medical device such as prosthetic implants or knee brace. There are many clinical applications for the knee joint that can be developed based on the IMU technology. For instance, the IMU can be used to monitor the progress of the patient undergoing physical therapy. Due to its compact size and simple set-up, it can be used in clinic as a substitution for gait lab for some of the diagnostic procedures. In addition, exercise science and sport science can use IMU as a feedback to the athletes in correcting their postures to improve their performances.

In this study, the IMU is used to monitor the orientation changes of the upper and lower extremities. Unlike the knee brace introduced in the pilot experiment, the IMUs are placed in separate container and strapped onto the subject's thigh and calf as shown in Figure 8-1. A passive optical tracker is securely attached to each container.

Two activities, deep knee bend (Figure 8-2) and chair rise (Figure 8-3), were performed under the surveillance of both IMU and optical tracking systems as shown. The following protocol was used for the data collection:

1. Subject stands still such that both of the optical trackers are facing the optical camera sensors and maintain the pose as steady as possible for 4000 samples. This is used as the reference pose for the activity.
2. The subject performs one continuous activity and pauses momentarily to facilitate data resynchronization during analysis
3. The subject repeats step 2 to acquire multiple dataset of the same activity.
4. At the end of each acquisition, the subject stands still for a brief period of time (~200 samples) before exiting the acquisition of the IMU and optical tracking system.

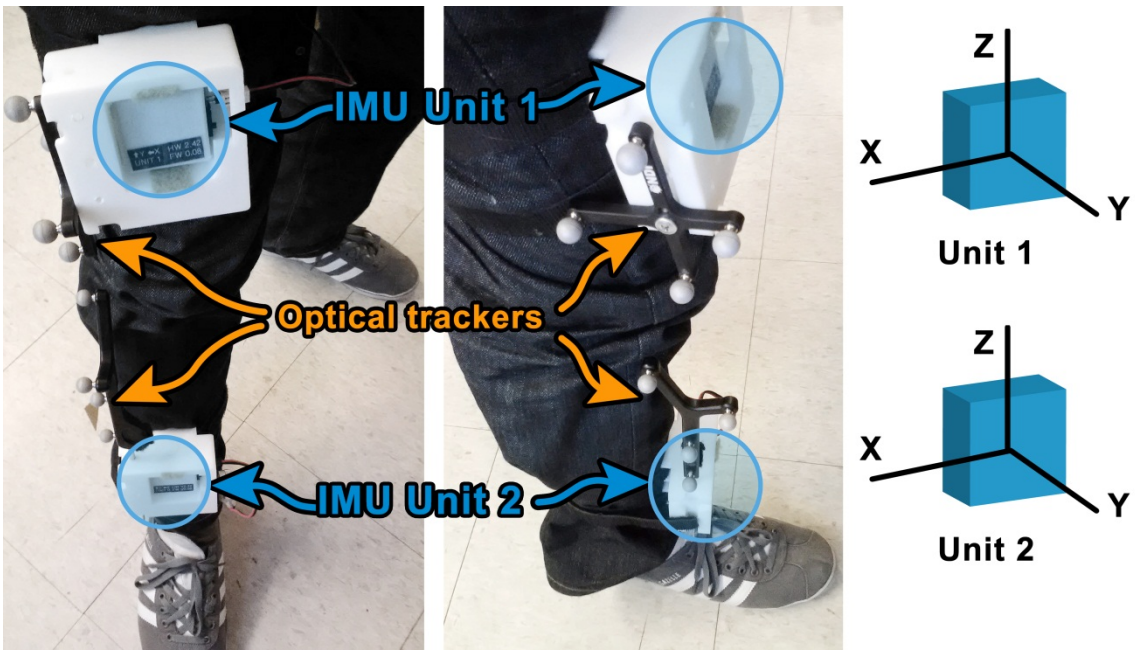

Figure 8-1 – Knee joint dynamic study set up. The coordinate system between the IMU tracker are shown in the left hand side of the figure.

Figure 8-2 –Subject performing deep knee bend activities

Figure 8-3 – Subject performaning chair rise activities

There are three sets of data collected for this experiment. Each set consists of multiple deep knee bend and chair rise activities. If the optical tracker is temporary outside of the viewing volume of the optical tracker during any of the activity, the missing data are interpolated. The collected data were processed by the algorithms in Table 7-1. The output quaternions are calculated relative to the reference quaternion established in step 1 of the protocol, which were then projected to 3-sphere via an identity matrix. The performance of the PF algorithm is compared to the benchmark algorithms relative to the optical data. The RMSE of all the testing algorithms were computed against the optical system. The outputs were averaged over all activity sets as shown in Figure 8-4 (thigh) and Figure 8-5 (shank). The average processing time in each recursion has

significantly increased due to the addition of a second IMU as shown in Figure 8-6. The red dashed line in the figure indicates the condition to achieve at least 30 fps. Most of the PFs failed to fulfill the condition except for the ones using vMF with particle population 256 or smaller and NU using AUX and BC with particle population of 256 or smaller.

The RMSE decreased slightly compared to the free hand motion experiment in previous chapter as illustrated by the numerical data in Table 8-1 and Table 8-2. This is due to the deep knee bend and chair rise activities have more constraints, where the majority of the motion revolves around one axis. The PF with AUX without BC remains to be the least accurate among all algorithms, while the PF using AUX and BC yields the best result. The RMSE also decreases as the particle population increases although the particle size to performance ratio is small. The best strategy with the consideration of accuracy and processing time is PF using NU density and BC with 256 particles.

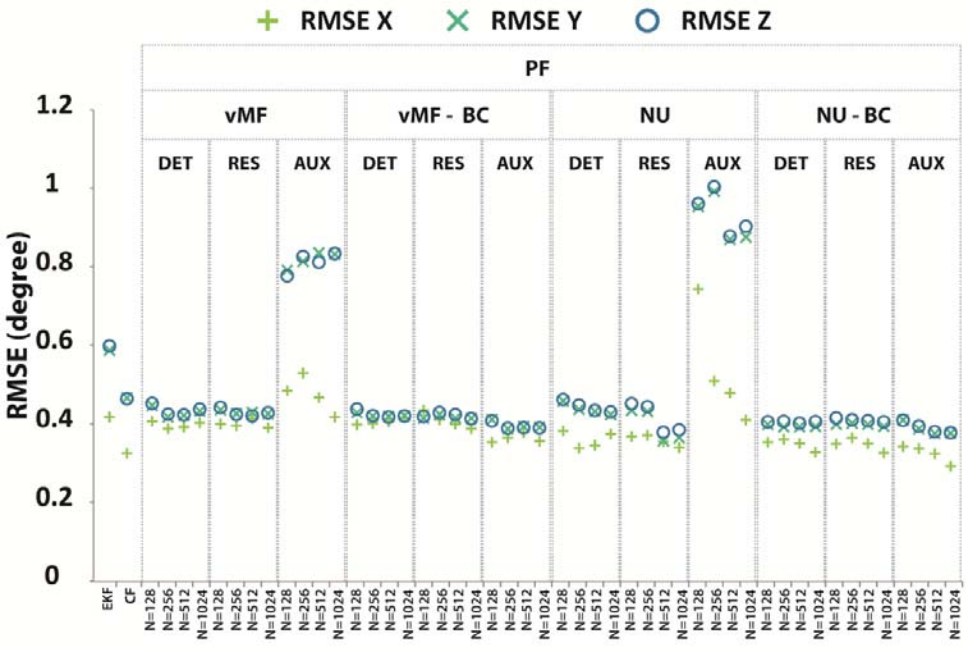

Figure 8-4 — Comparison between the RMSE of PF and the benchmark algorithms for IMU located at thigh during knee joint activities

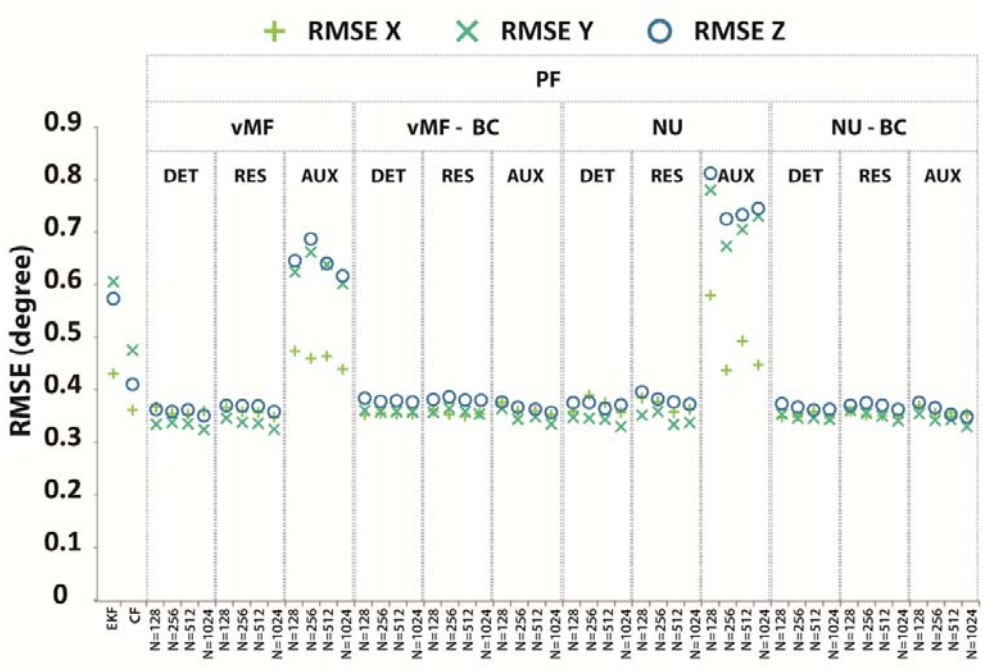

Figure 8-5 — Comparison between the RMSE of PF and the benchmark algorithms for shank during knee joint activities

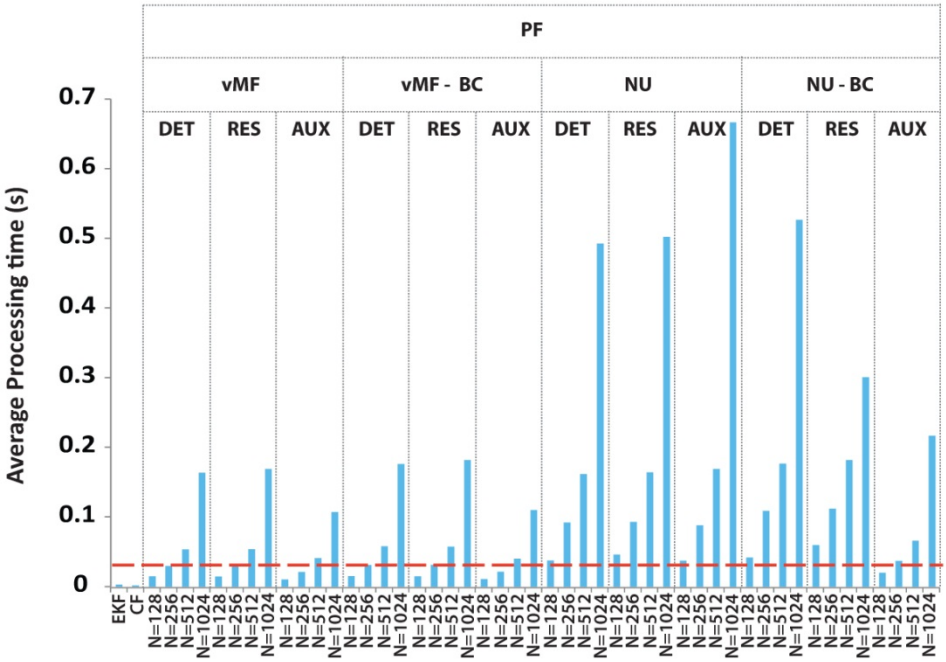

Figure 8-6 — Average processing time of all testing algorithms for knee joint activities

Table 8-1 - RMSE of the testing algorithms for dynamic testing (thigh)

Algorithm				RMSE X	RMSE Y	RMSE Z
CF				0.324628	0.465855	0.463407
EKF				0.416998	0.587236	0.598235
	Posterior Density	Importance Resampling	Particle size			
PF	vMF	DET	128	0.40652	0.446681	0.451516
			256	0.387251	0.417212	0.424143
			512	0.391946	0.421225	0.422965
			1024	0.402673	0.432454	0.436764
		RES	128	0.39931	0.435609	0.441193
			256	0.394968	0.426112	0.423514
			512	0.413259	0.429296	0.420109
			1024	0.390083	0.425742	0.427685
		AUX	128	0.484144	0.79161	0.777059
			256	0.529528	0.814311	0.826135
			512	0.466454	0.835663	0.811539
			1024	0.416824	0.832426	0.833625
	vMF BC	DET	128	0.397709	0.428169	0.436941
			256	0.400959	0.419716	0.420432
			512	0.404553	0.418835	0.416868
			1024	0.415252	0.41994	0.419715
		RES	128	0.435045	0.414229	0.418314
			256	0.410515	0.423846	0.428496
			512	0.398688	0.414318	0.423451
			1024	0.387155	0.413804	0.412882
		AUX	128	0.352875	0.410506	0.40724
			256	0.364042	0.386864	0.388586
			512	0.377671	0.392906	0.389772
			1024	0.355447	0.391104	0.389053
	NU	DET	128	0.381415	0.457865	0.461421
			256	0.337517	0.436777	0.447388
			512	0.345554	0.431794	0.433812
			1024	0.373989	0.423383	0.430074
		RES	128	0.367673	0.434364	0.450963
			256	0.370254	0.431317	0.443187
			512	0.354173	0.356792	0.378091
			1024	0.339651	0.36424	0.38462
		AUX	128	0.742855	0.953975	0.961032
			256	0.508693	0.99229	1.00357
			512	0.479046	0.870522	0.877844
			1024	0.409379	0.875489	0.903102
	NU BC	DET	128	0.368647	0.522497	0.443228
			256	0.371336	0.509478	0.443086
			512	0.36801	0.521242	0.442473
			1024	0.351464	0.518449	0.445021
		RES	128	0.368075	0.52139	0.451569
			256	0.38154	0.52649	0.448103
			512	0.366057	0.51908	0.446344
			1024	0.349975	0.514702	0.44233
		AUX	128	0.352392	0.504013	0.440037
			256	0.339917	0.486217	0.430011
			512	0.322679	0.473601	0.416804
			1024	0.29787	0.469611	0.412399

Table 8-2 - RMSE of the testing algorithms for dynamic testing (shank)

Algorithm				RMSE X	RMSE Y	RMSE Z
CF				0.361234	0.475167	0.410433
EKF				0.430701	0.605816	0.573356
	Posterior Density	Importance Resampling	Particle size			
PF	vMF	DET	128	0.36697	0.33451	0.362384
			256	0.355031	0.337851	0.35836
			512	0.353171	0.335397	0.361793
			1024	0.360111	0.324276	0.350328
		RES	128	0.3663	0.34629	0.36993
			256	0.358331	0.33823	0.369685
			512	0.356747	0.336699	0.368974
			1024	0.347879	0.324751	0.35846
		AUX	128	0.473697	0.624209	0.645817
			256	0.460228	0.662193	0.687106
			512	0.464368	0.638276	0.640524
			1024	0.439086	0.601871	0.616599
	vMF BC	DET	128	0.351879	0.360873	0.383911
			256	0.355727	0.358138	0.377867
			512	0.353474	0.35858	0.379108
			1024	0.352698	0.35758	0.376628
		RES	128	0.359841	0.357472	0.381631
			256	0.352841	0.365049	0.386456
			512	0.35049	0.358687	0.380419
			1024	0.354917	0.353847	0.380386
		AUX	128	0.376686	0.364286	0.376351
			256	0.359598	0.343834	0.366673
			512	0.36004	0.348186	0.363026
			1024	0.352518	0.335008	0.356943
	NU	DET	128	0.358632	0.347195	0.375258
			256	0.390247	0.346124	0.375843
			512	0.37515	0.343721	0.364373
			1024	0.357094	0.330081	0.370834
		RES	128	0.384424	0.351709	0.396317
			256	0.377658	0.358932	0.382354
			512	0.357676	0.334205	0.376862
			1024	0.362837	0.33792	0.372338
		AUX	128	0.579639	0.78012	0.81227
			256	0.437536	0.673492	0.725298
			512	0.493264	0.705367	0.733129
			1024	0.44728	0.730884	0.744864
	NU BC	DET	128	0.347802	0.353396	0.373244
			256	0.349527	0.346308	0.366912
			512	0.358042	0.346313	0.361706
			1024	0.346902	0.343523	0.36292
		RES	128	0.359612	0.360545	0.370087
			256	0.351709	0.35519	0.375252
			512	0.35106	0.350104	0.37031
			1024	0.3465	0.341601	0.362705
		AUX	128	0.369503	0.354304	0.375568
			256	0.355348	0.342114	0.365768
			512	0.356148	0.343175	0.35282
			1024	0.353763	0.330291	0.348451

8.2 Lumbar spine activities

In United States, low back pain is one of the most common musculoskeletal disorders that contribute to at least 50 billion dollars in health care cost annually. [106]. The cause of low back pain varies greatly, which can be from simple muscle sprain to degenerative discs. Physicians rely heavily on magnetic resonance imaging (MRI) in diagnosing low back pain, which is the leading cost to health care expenses. In a recent study, fluoroscopy analysis was performed on normal, normal with low back pain, and disc degeneration subjects in several activities. The hypothesis for this phenomenon was that the kinematics of the degenerative subjects has changed to avoid pain, which can be observed as out of plane motions during these activities. The study developed a classification algorithm that differentiates the normal and degenerative subjects based on the kinematics of the L1 and L5 in the lumbar region [107]. The ability to differentiate normal and degenerative patients with kinematics information during diagnosis can reduce the instance requiring MRI.

In this study, IMUs were used to demonstrate the capability as a preliminary diagnostic device to obtain the kinematics information of L1 and L5 of the subjects. There were two subjects participating in this experiment. One of the subjects (Subject 1) is classified as normal and the other (Subject 2) is classified as degenerative according to previous study [107]. The IMU is secured in positions on the back of the subject with Tegaderm (3M) as shown in Figure 8-7. Optical tracker was not used in this experiment due to the size of the trackers are too large and they interfere with the motions of the subject. The subjects were asked to perform three activities, including flexion/extension (Figure 8-8), lateral bending (Figure 8-9) and axial rotation (Figure 8-10). The following protocol was used for lumbar activity data collection:

1. Subject stands still for a duration of 4000 samples to establish the reference positions
2. Subject performs multiple flexion and extension activities
3. Subject performs multiple lateral bending activities
4. Subject performs multiple axial rotation activities
5. At the end of each acquisition, the subject stands still for a brief period of time (~200 samples) before exiting the acquisition of the IMU tracking system.

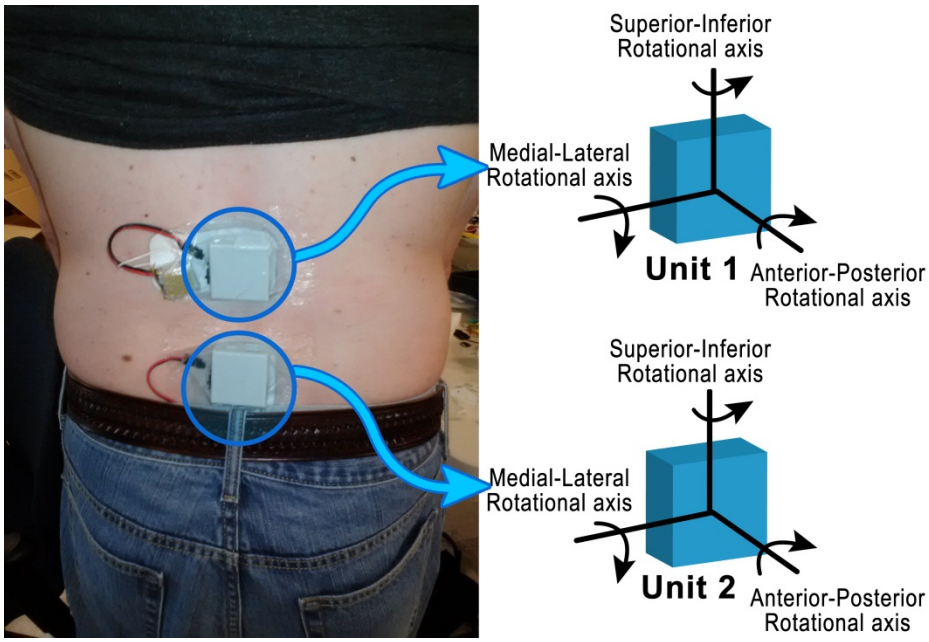

Figure 8-7 – IMUs setup for lumbar spine experiment

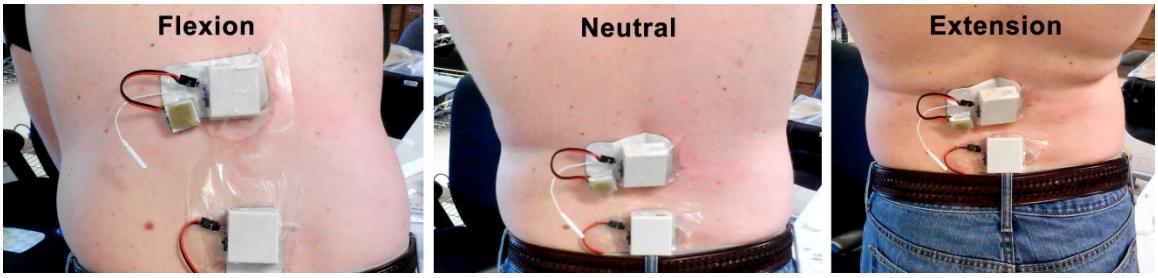

Figure 8-8 – Subject performing flexion / extension activity

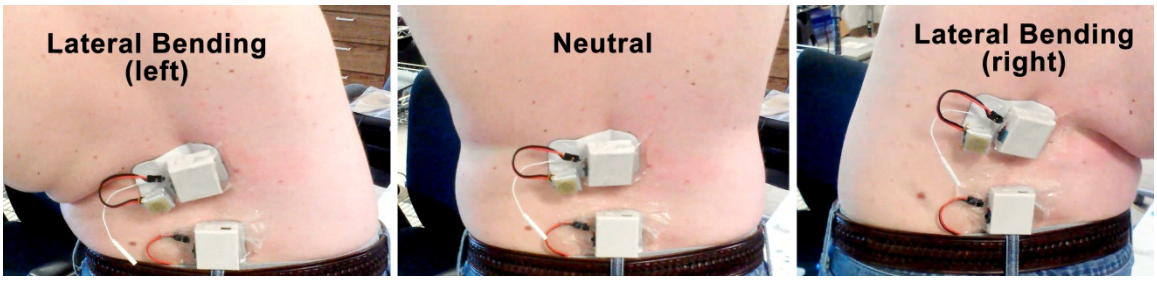

Figure 8-9 – Subject performing lateral bending activity

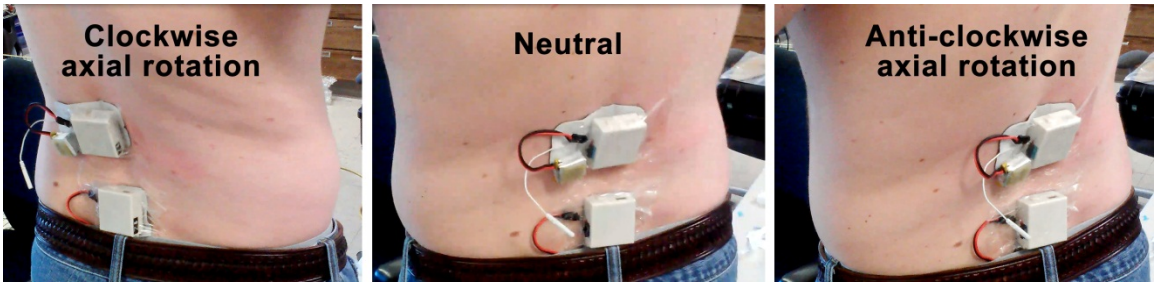

Figure 8-10 – Subject performing axial rotation activity

Based on the result from previous experiments, PF algorithm using NU density and BC with 1024 particles was chosen to process the data. Figure 8-11 and Figure 8-12 shows the outputs on each rotational axis of the IMU located at the L1 region for healthy and degenerative subject respectively. Figure 8-13 and Figure 8-14 shows the outputs on each rotational axis of the IMU located at the L1 region for healthy and degenerative subject respectively. The definition of the rotational axis is shown in Figure 8-7. During the experiment with the degenerative subject, the IMU at L5 was temporary slipped out of placed during the flexion / extension activity. A new reference position was re-defined during the temporary pause after the flexion / extension activity. All subsequent data were referenced to this new orientation in the analysis. Figure 8-15 shows the procedure to calculate the relative angles between the two IMUs. The quaternion is projected with three orthogonal vectors. The relative angles between the vectors in L1 and L5 were calculated by the dot product between the vectors. The result gives the absolute angle of each axis of the L1 IMU in the L5 IMU frame.

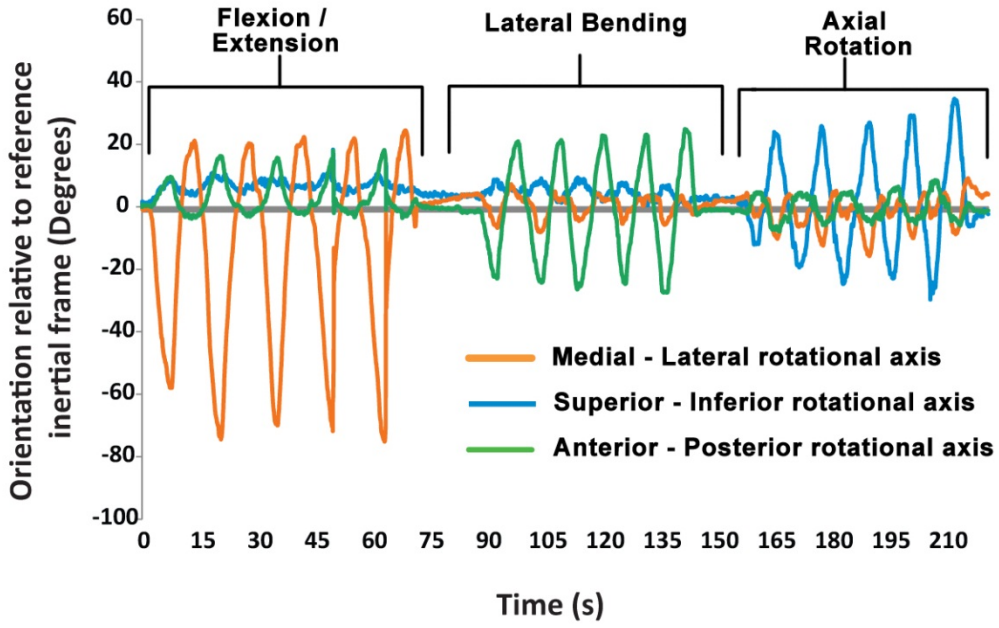

Figure 8-11 – Orientations of IMU located at L1 relative to the reference inertial frame for healthy subject

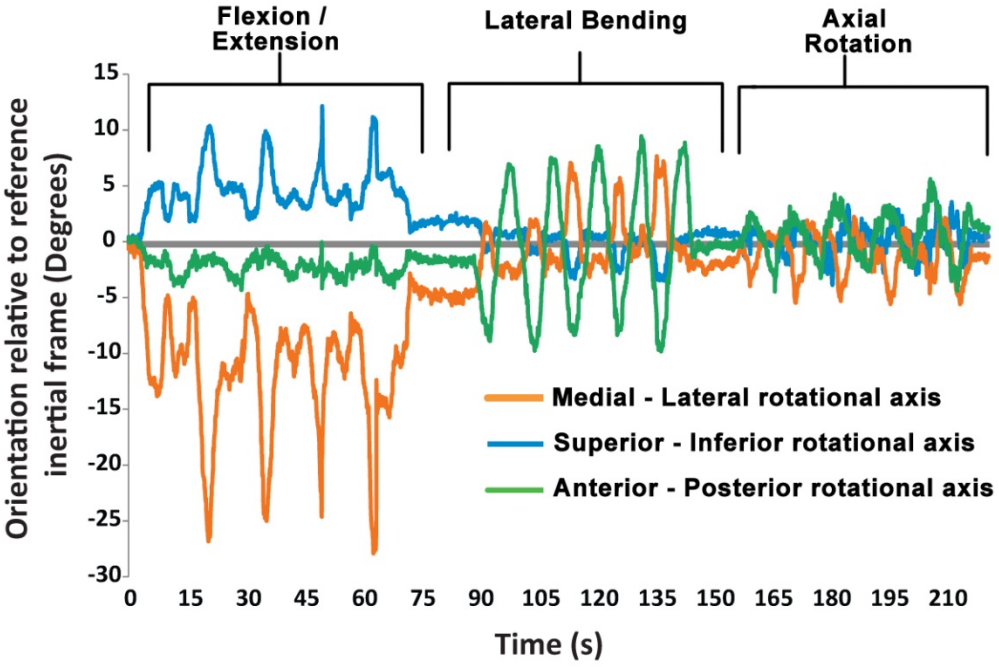

Figure 8-12 – Orientations of IMU located at at L5 relative to the reference inertial frame for healthy subject

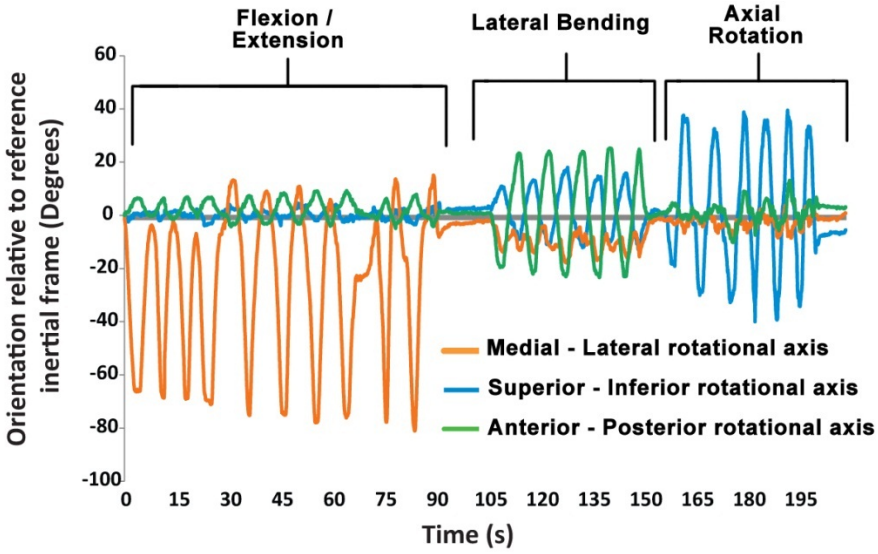

Figure 8-13 – Orientations of IMU located at L1 relative to the reference inertial frame for degenerative subject

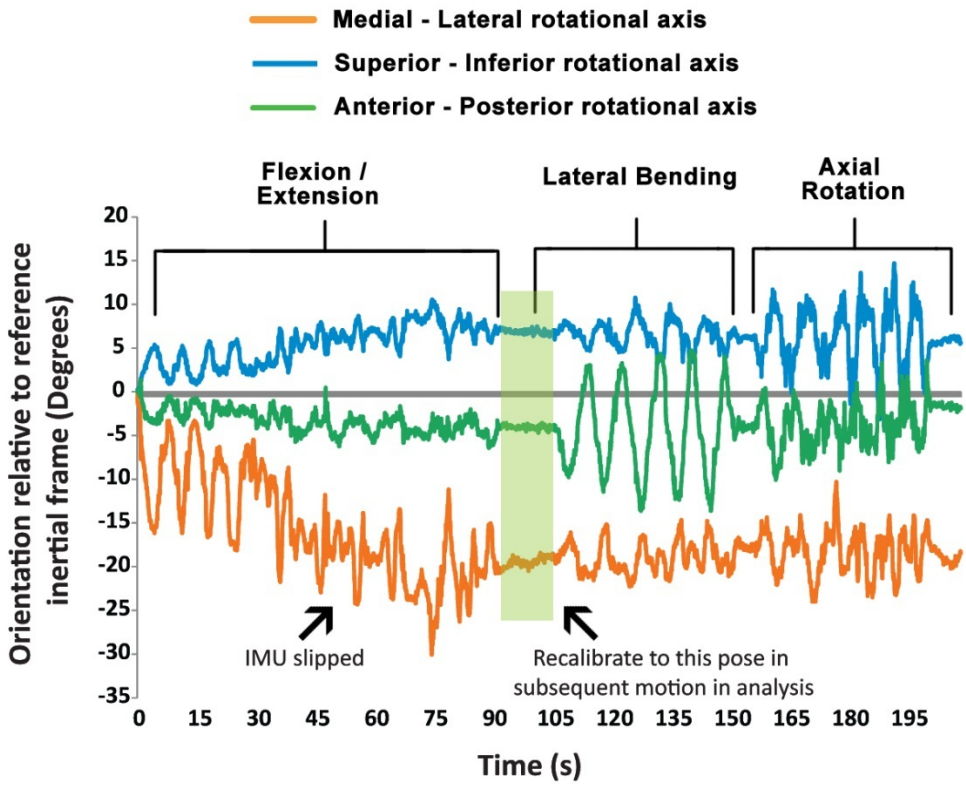

Figure 8-14 – Orientations of IMU located at L5 relative to the reference inertial frame for degenerative subject

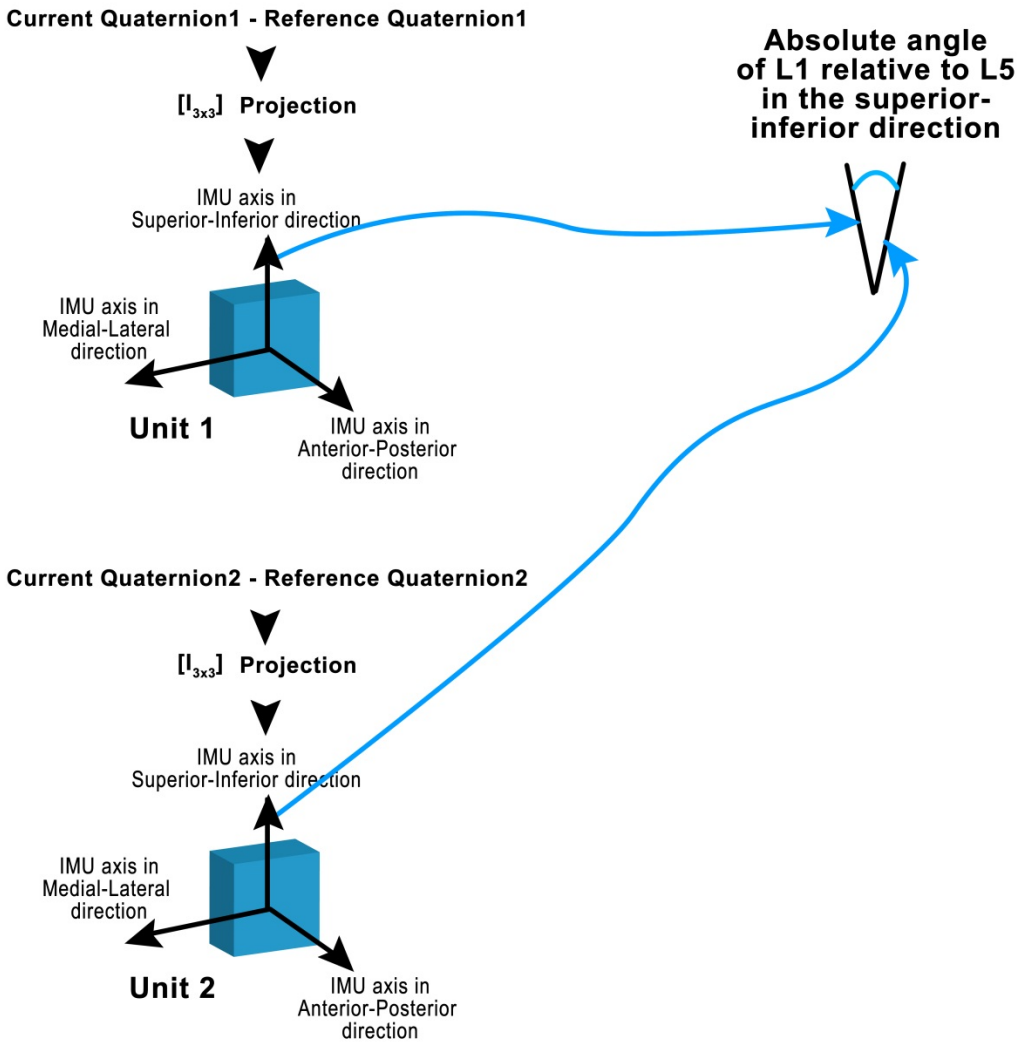

Figure 8-15 – Method to calculate the absolute orientation difference between the L1 IMU and L5 IMU axis in the L5 IMU frame

The absolute change in orientation between L1 IMU and L5 IMU axis during flexion/extension activity (Figure 8-16 and Figure 8-17), lateral bending (Figure 8-18 and Figure 8-19), and axial rotation (Figure 8-20 and Figure 8-21) for healthy and degenerative subjects were analyzed. Out of plane motion is characterized from observing the two axes perpendicular to the rotational changes. For instance, during the flexion/extension activity, the IMU axes in the anterior-posterior (AP) and superior-inferior (SI) directions were rotated around the medial-lateral (ML) rotational axis. Hence, the absolute angular change of these axes should be very similar to each other. However, an orientation offset between the two axes indicates out of plane motion between the IMUs on L1 and L5. In addition, the result shows that out of plane motion can be detected much easier during the lateral bending activity.

During the flexion/extension activity shown in Figure 8-16 and Figure 8-17, the ratio between the in-plane activity motion and out-of plane motion is so large that it is difficult to identify abnormal motion. However, during the lateral bending activity shown in Figure 8-18 and Figure 8-19, out of plane motion can be identified easily on the degenerative subject during the activity. The relative angular difference between the IMU axis in L1 and L5 in the ML and SI directions are similar with the healthy subject, while there is up to 10 degrees of orientation offset between the two axes for the degenerative subject in some instances. Similar phenomenon can also be observed during the axial rotation activity shown in Figure 8-20 and Figure 8-21. The axial rotation activity is less preferable than lateral bending activity because the back muscles of the subject bulges significantly during axial rotation, which push and slide the IMU away from the anchored position.

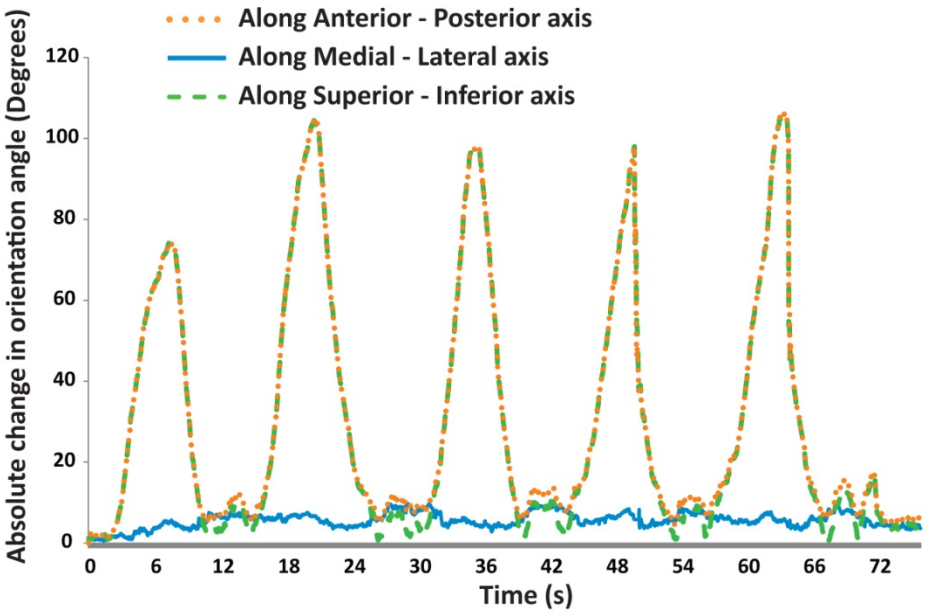

Figure 8-16 – Absolute change in orientation of each axis of L1 relative to L5 for healthy subject during flexion /extension activities

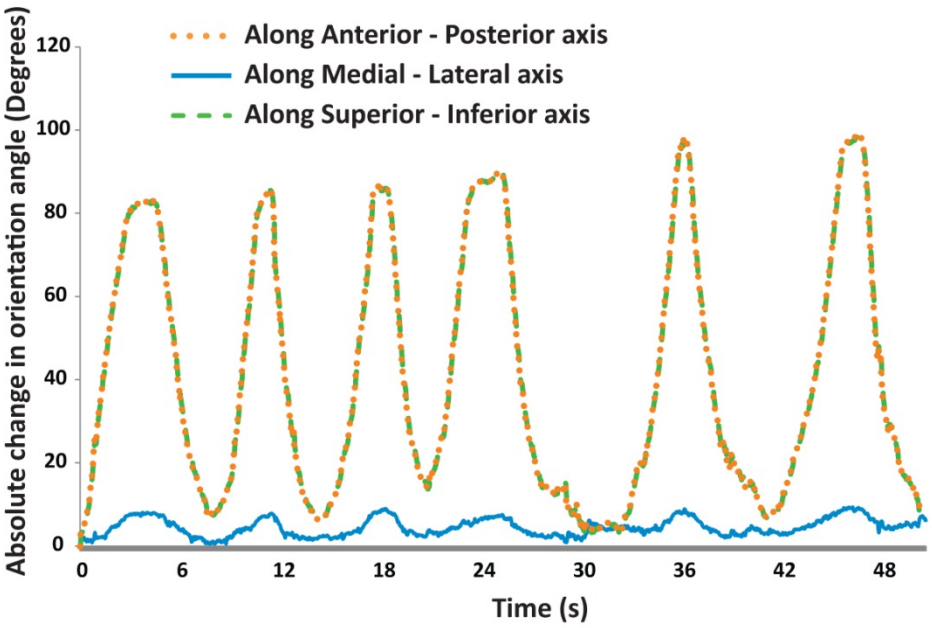

Figure 8-17 – Absolute change in orientation of each axis of L1 relative to L5 for degenerative subject during flexion /extension activities

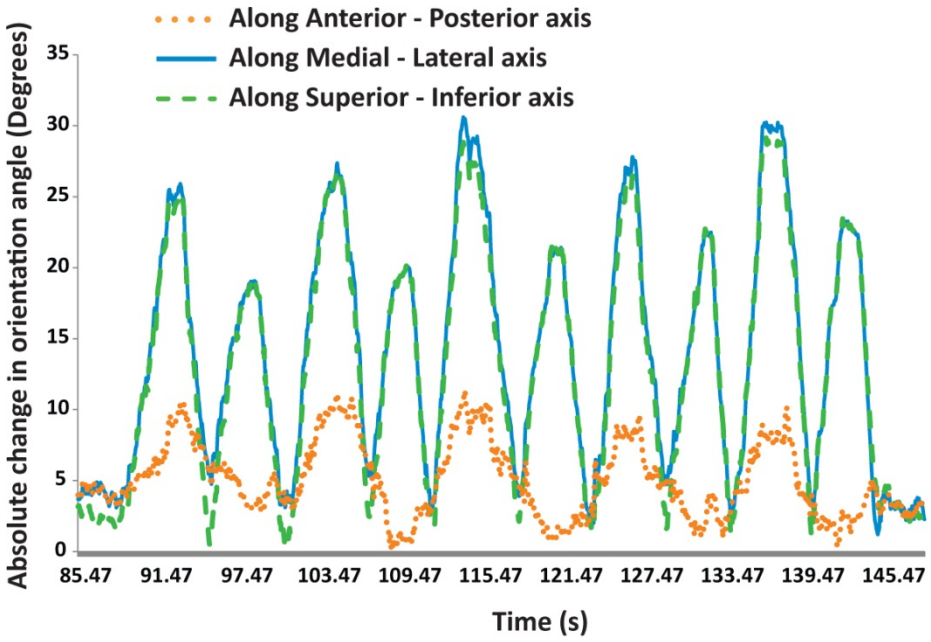

Figure 8-18 – Absolute change in orientation of each axis of L1 relative to L5 for healthy subject during lateral bending activities

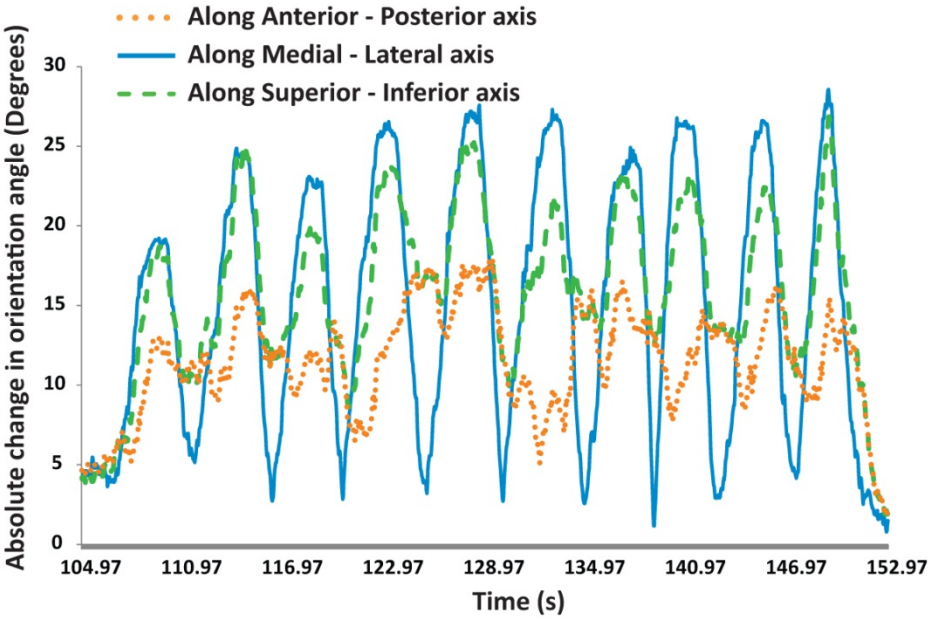

Figure 8-19 – Absolute change in orientation of each axis of L1 relative to L5 for degenerative subject during lateral bending activities

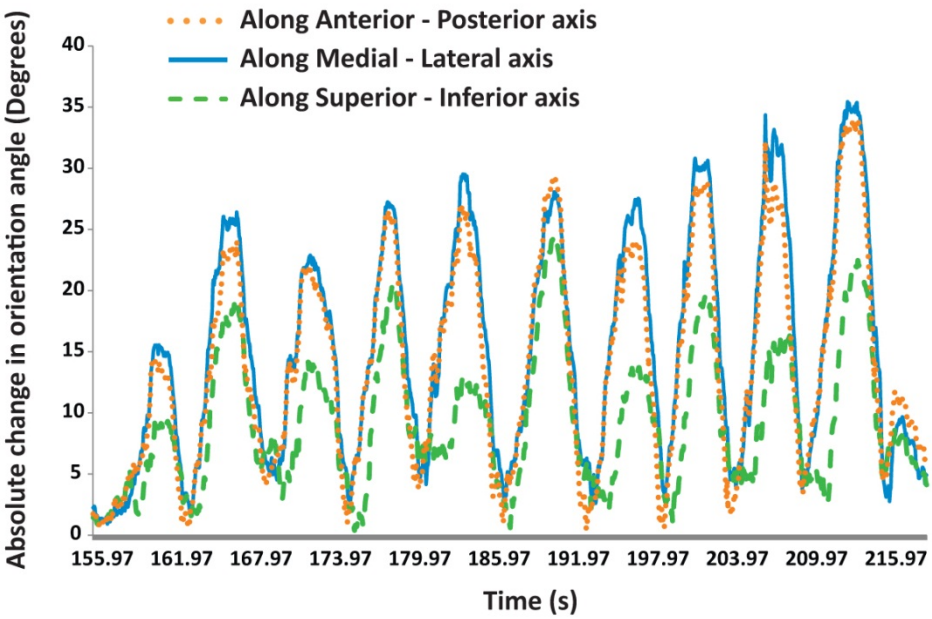

Figure 8-20 – Absolute change in orientation of each axis of L1 relative to L5 for healthy subject during axial rotation activities

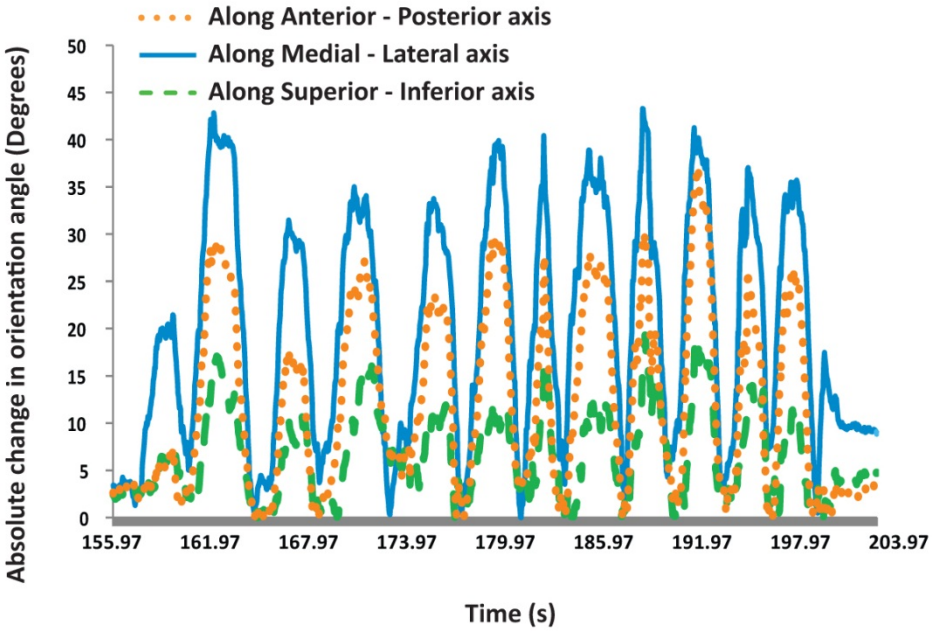

Figure 8-21 – Absolute change in orientation of each axis of L1 relative to L5 for healthy subject during axial rotation activities

The primary issue with current IMU configuration for lumbar assessment is that the container of the IMU does not fit very well with the back of the subject as shown in Figure 8-22. This creates unwanted spaces for the IMU to slide during the activity. A different ergonomic design to house the IMU tracking system can be developed to remedy this problem. Nevertheless, this experiment demonstrates the potential use of IMU orientation tracking to differentiate normal and degenerative patients.

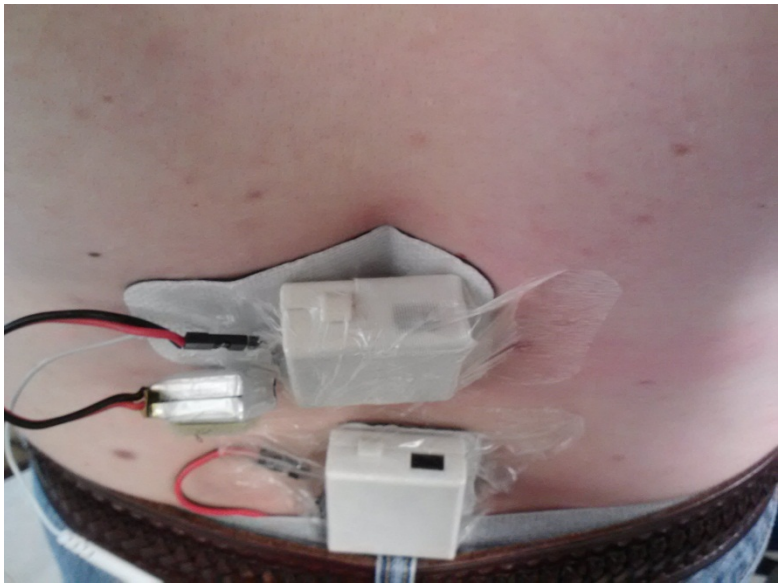

Figure 8-22 – Empty space between the IMU and the back of the subject

9. Conclusion

This dissertation provides an extensive investigation of inertial tracking technologies for human motion monitoring. There are three important factors that affect the accuracy of an inertial tracking system, which are hardware design, calibration and algorithm design. This dissertation examined and optimized all three factors through novel designs and approaches.

The calibrations of the IMU sensors were discussed in details in Chapter 4. The current primary cause of error in estimation rooted from the cross axis effect from the magnetometers. The cross axis effect compensation method was discussed in Chapter 7. As discussed in Chapter 5, the modular design of the IMU allows flexibility to the tracking system, which allows fast customization of the system that best suited the dynamics of the activities. This design has also taken account into the ADC dynamic range and performance degradation at the maximum convention speed to provide high performance and high resolution system. Chapter 2 provides extensive review on the orientation representations and the theoretical background to the tracking problem, which leads to the conclusion of the necessity of using quaternion with sequential Monte Carlo method to tackle the human motion tracking problem. These theories for attitude estimation were realized and discussed in details in Chapter 6. In addition, a novel approach for quaternionic particle filtering for tracking problem is realized by introducing hyperdimensional directional statistic geometries. The sampling methods for these statistical geometries, the von Mises-Fisher and Bingham densities, were discussed. A non-uniform density is also introduced by modifying the sampling method of the Bingham density. A fast and novel implementation to calculate weighted quaternions were also demonstrated. Lastly, a novel 2-stage resampling techniques were designed for each of the statistical distribution to maintain the posterior density of the

estimation while increasing the particle diversity and preventing density divergence from faulty sensors input.

An extensive validation study for the IMU and the attitude estimation algorithms was discussed in Chapter 7. A synthetic signals model is introduced to simulate the noisy outputs from the sensors. It is also demonstrated that Gaussian noise from individual system can lead to non-Gaussian system due to the data fusion technique. The system is tested with robotic manipulator as well as human free hand motion. An optical system serves as the ground truth and is used as the bases to compare with the benchmark algorithms, which are the Extended Kalman Filter and the Complementary Filter. The validation study also demonstrates the hardware upgrade on the IMU improves the estimation significantly compare to the off the shelf IMU in Chapter 4. It is also shown that the particle filter performs slightly better than the benchmark algorithms. However, the primary advantage of the particle filtering over the benchmark algorithm is stability. The interaction between IMU and the musculoskeletal system invalidates some of the assumptions in the complementary filter designs, which prevent the filter to converge to a solution. The instability of the complementary filter during free hand motion is found to be statistically significant (Fisher exact test, $p < 0.05$).

Chapter 8 discusses the biomedical applications of the IMU orientation tracking. The first study demonstrates the capability of tracking knee joint during activities and the accuracy was accessed by comparing with optical tracker. In the second study, the IMU tracking system was used on monitoring motion for normal and degenerative subjects. The study demonstrates the potential of using IMU as a diagnostic device that differentiate normal patient with low back pain to degenerative patients.

10. Future work

The design of a highly accurate inertial tracking system is a multi-discipline optimization problem. There are several areas that can be improved in the current system. The following sections discuss the future work in the hardware design, calibration techniques, algorithms for this system, and potential applications, which is also summarized in Table 10-1. In addition, potential applications for using inertial tracking system for biomedical applications are discussed.

10.1 Hardware

There are several hardware upgrades that can improve the performance of the current system. The most important one is the wireless transmission system. Since the current IMU firmware simply instructs the transmitter to transmit the data whenever a packet becomes available in the buffer of the microcontroller, the receiver may receive multiple packets from one IMU before receiving the data from the second unit. In addition, the receiver's firmware has to identify the origin of the received data prior to type-casting the data to send to the computer. In the current setup, the wireless transmission protocols discarded approximately 25% of the receiving data. This occasionally introduces time lag in the data. Since the expectation of the attitude estimation is rate dependent, time lag introduces error in the prediction. Even though the error is corrected once new observation data is obtained, the rubber-banding effect is still undesirable. This can be remedied with using high capacity wireless protocol or UWB transmission that supports high data rate as well as upgrading the firmware of the receiver by implementing time division multi-access (TDMA) scheme on the receiver.

Secondly, the current option for sensing strips is still limited. The selection can support most of the normal daily activities. Expanding the sensor selections will allow testing on more rigorous activities such as running.

Thirdly, other sensor types can be introduced into the system to perform other tracking purposes. Ultrasonic or infrared range finder can provide translational information that is currently unobtainable. With the translation data, it is possible to perform Simultaneous Localization And Mapping (SLAM) with medical ultrasound transducer to map and create the tissues model (e.g. bone).

Lastly, as demonstrated in the current implementation of the particle filter, the bottleneck of the algorithm is the processing time. While it is feasible to process and display the data in real time for single IMU, there is a heavy burden of the processing for each additional IMU joining the network. One of the possible solutions is to shift the burden of data processing onto the IMU system. Since microcontroller does not have the capacity to process complicated algorithm such as particle filter, which requires exhaustive memory, programmable logic device such as field programmable gate array (FPGA) becomes a good candidate for data processing for the IMU.

10.2 Calibration

As mentioned in previous chapter, magnetometer calibration is severely complicated due to the cross axis effect. The cross axis coefficients must be determined in each location and preferable every time prior to data acquisition. The future work in this area will included a mechanical gimbal that rotates the IMU in multiple orientations. An optimization algorithm can be used to automate the coefficients estimation.

10.3 Algorithm Design

The current particle filter algorithm provides an initial framework for quaternionic attitude estimation. There are vast variety of distributions and sampling method that is yet to explore. The primary focus is to design an algorithm that allows

fast sampling and evaluation. There are many importance resampling methods that have to be examined. In addition, the current algorithm has not fully utilized the capability of the IMU system. The modular IMU uses sensors of the same kind but with different configurations or has different dynamic range. However, since they are monitoring the same motion, an optimization algorithm can be used with different configuration to compute the optimal results.

10.4 Potential Applications

Besides of the applications introduced in the chapter 8, there are many other potential applications to use IMU orientation tracking. From the perspective of joint motion tracking, once the characteristic of the activity such as the range of motion and speed are identified, the systems can be used reconfigured with appropriate sensing elements that suit the needs of the application. The challenging aspect, however, is designing an ergonomic container for the tracker such that it can be mounted easily onto the joints without hindering or altering the range of motion of the joint. In addition, multi-joint segments tracking are also possible with combination of more than two IMU systems. However, the computation burden of the particle filter in each IMU must be elevated before realizing this system.

One of the emerging technologies is computer assisted surgeries. Many new surgical techniques use computer assisted surgical planning in the effort to minimize surgery time and improve the overall performance. During the surgery, optical system is frequently used to track and provide position feedback of the instruments according to the surgical plans. Since IMU cannot be used to track translational movement, a tightly coupled IMU and UWB localization system can potentially replace the optical tracking

system. The advantage of that is that neither system has line-of-sight requirements, which allows the surgeons to move around more freely during the surgery.

Another application is to couple the IMU-UWB localization system with ultrasound transducers for in-vivo bone motion monitoring. Currently, there are limited options for performing in-vivo kinematics analysis. Most of the in-vivo examinations are done with either X-ray or fluoroscopy. Future work in this are included using IMU-UWB localization for external positing observation, and the ultrasound system monitors the skin to bone correction. The complete system can estimate the in-vivo kinematics of the joint motion.

Table 10-1 – Summary on future work

Hardware	Upgrade wireless transmitter that supports higher data rate Wireless transmission protocol More variety of sensing strips Integration with other sensing system Shift computational burden to FPGA
Calibration	Cross axis compensation protocol Optimization algorithm to estimation cross axis coefficient
Algorithm	Faster sampling and evaluation Improvement on resampling method Optimization algorithm based on the modular architecture
Application	Multi-joint segment tracking Tightly coupled UWB-IMU system for computed assisted surgery Tightly coupled UWB-IMU-US system to perform SLAM on bone tissues Tightly coupled UWB-IMU-US system to perform in-vivo joint motion analysis

REFERENCES

[1] Wirth H. Dorfmuller K., "Real-time hand and head tracking for virtual environments using infrared beacons," *Proceedings CAPTECH 98*, vol. 1537, pp. 113-127, 1998.

[2] Cai Q. Aggarwal J.K., "Human Motion Analysis: A Reivew," *IEEE Nonrigid and Articulated Motion Workshop*, pp. 90-102, 1997.

[3] Kaufman K.R., "Future direction in gait analysis," *RRDS Gait Analysis in the Science and rehabilitation*, vol. 4, pp. 85-112, 1998.

[4] Thompson D., Frantz D. Wiles A., "Accuracy assessment and interpretation for optical tracking systems," *Proc.of SPIE*, vol. 5367, pp. 421-432, 2004.

[5] Northern Digital Inc. (2008) Polaris Spectra & Polaris Vicra technical specifications. [Online]. http://www.ndigital.com/medical/polarisfamily-techspecs.php

[6] Motion Analysis Corporation. Motion Analysis Corporation Raptor E Digital Real Time system. [Online]. http://www.motionanalysis.com/pdf/RaptorE_System.pdf

[7] Raaf F., Blood E., Steioner T., and Jones H., "Magnetic position and orientation tracking system," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 5, no. 15, pp. 709-718, 1979.

[8] Nixon M, McCalium B, Frigh W, and Price B, "The effects of metals and interfering fields on electromagnetic trackers," *Presence*, vol. 7, no. 2, pp. 204-218, 1998.

[9] Animian K, "Monitoring human movement with body-fixed sensors and its clinical application," *Computational Intelligence for Movement Sciences*, pp. 101-138, 2006.

[10] Best R. and Begg R., "Overview of Movement Analysis and Gait Features," *Computational Intelligence for Movement Sciences*, pp. 17-69, 2006.

[11] Kindratenko V.V., "Calibration of Electromagnetic Tracking Devices," *Virtual Reality*, vol. 4, pp. 139-150, 1999.

[12] Ascension Technologies Corporation. (2008) Flock of Birds, real-time motion tracking. [Online]. http://www.ascension-tech.com/docs/Flock_of_Birds.pdf

[13] Hofmann-Wellenhof B., Lichtenegger H., and Collins J., *Global Positioning System. Theory and Practice*.: Springer-Verlag Ltd., 1993.

[14] Klobuchar J.A., "Ionospheric Effects on GPS," Air Force Georphysic Laboratory, GPS World, 1991.

[15] Collins J.P., Langley R.B. Mendes V.B., "The effect of Tropospheric Propagation Delay Error in Airborne GPS Precision Positioning," in *8th International Technical Meeting of Satelite Division of the Institute of Navigation*, Palm Spring, CA, September, 1995.

[16] Zhang C., Merkl B.C., Kuhn M.J., Fathy A.E. Mahfouz M., "Investigation of High Accuracy Indoor 3-D Positioning Using UWB Technology," *IEEE Transaction on Microwave Theory and Techniques*, vol. 56, no. 6, pp. 1316-1330, 2008.

[17] M. Kuhn, Zhang C., Mahfouz M., Merkl B., and Fathy A., "A system-level simulation framework for UWB localization," *IEEE Trans on Microwave Theory Tech.*, vol. 58, no. 12, pp. 3527-3537, December 2010.

[18] Zhang C, Kuhn M., Fathy A., and Mahfouz M., "Real-time noncoherent UWB positioning radar with millimeter range accuracy in a 3D indoor environment," *IEEE Trans. on Microwave Theory and Tech.*, vol. 58, no. 1, pp. 9-20, January 2010.

[19] NASA (FS-2001-03-017-GSFC) , Robert H. Goddard: American Rocket Pioneer.

[20] Draper C.S. et al., "Space Navigation Guidance and Control," vol. 1, 1965.

[21] Xsens Coporation. MTx specification. [Online]. http://www.xsens.com/images/stories/products/PDF_Brochures/mtx%20leaflet%2009.pdf

[22] MicroStrain Corporation. Inertia-Link specification. [Online]. http://www.microstrain.com/pdf/Inertia-Link%20datasheet%20-%20v%202.pdf

[23] Spiegelmann R. and Harnof S., Surgical Navigation with BrainLAB System, 2003, Informa Healthcare.

[24] Lorakul A. and Suthakorn J., "Towards Robot-Assisted Dental Surgery: Path Generation and Navigation System Using Optical Tracking Approach," in *IEEE Conference on Robotics and Biomimetics*, Bankok, Thailand, February 2009.

[25] Stiehl J.B., Dennis D.A., and R.D. Komistek, "In Vivo Kinematic Analysis of a Mobile Bearing Total Knee Prosthesis," *Clinical Orthopedics & Related Research*, no. 345, pp. 60-66, 1997.

[26] Mahfouz M.R., Hoff W.A., and Komistek R.D., "A robust method for registration of three-dimensional knee implant models to two-dimensional fluoroscopy images," *IEEE Transaction on Medical Imaging*, vol. 12, no. 2, pp. 1564-1574, 2003.

[27] Komistek R.D., Dennis D.A., and Mahfouz M.R., "In Vivo Fluoroscopic Analysis of the Normal Human Knee," *Clinical Orthopedics & Related Research*, no. 410, pp. 69-81, May 2003.

[28] Davidson J.M., "Roentgen rays and localization. An apparatus for exact measurement and localization by means of roentgen rays," *Journal of British Mediciine*, 1898.

[29] Selvil G., *A roentgen-stereophotogrammetric method for the study of kinematics of the skeletal system*. Lund Sweden: University of Lund, 1974, Thesis.

[30] Fuller J., Liu. L.J., Murphy M.C., and R.W. Mann, "A comparison of lower-extremity skeletal kinematic measured using skin- and pin-mounted markers," *Human Movement Science*, vol. 16, pp. 210-242, 1997.

[31] Kaptein B.L., E.R. Valstar, B.C. Stoel, P.M. Rozing, and J.H. Reiber, "A new model-based RSA method validated using CAD models and models from reversed engineering," *Journal of Biomechanics*, vol. 36, no. 6, pp. 873-882, 2003.

[32] Veltink P.H. and Boom H.B.K., "3D movement analysis using accelerometry theoretical concepts," in *Neuroprosthetics From Basic Research to Clinical Applications*, Pedotti A. et al., Eds.: Springer; Berlin, 1996, pp. 317-326.

[33] Bouten C.V.C., Koekkoek K.T.M., Verduin M., and Kodde R., "A Triaxial Accelerometer and Portable Data Processing Unit for the Assessment of Daily Physical Activity," *IEEE*

Transaction of Biomedical Engineering, vol. 44, pp. 136-147.

[34] Forester F., Smejia M., and Fahrenberg J.U., "Detection of posture and motion by accelerometry: a validation study in ambulatory monitoring," *Comput. Human Behav.*, vol. 15, no. 151-183, 1999.

[35] Mathe M.J., Coster A.C., Lovell N.H., and Celler B.G., "Detection of daily physical activities using triaxial accelerometer," *Med. Biol. Eng. Comput.*, vol. 41, pp. 296-301, 2003.

[36] Uswatte G., "Objective measurement of functional upper extremity movement using accelerometer recordings transformed with a threshold filter," *Stroke*, vol. 31, pp. 662-667, 2000.

[37] Miyazaki S., "Long term unrestrained measurement of stride length and walking velocity utilizing a piezoelectric gyroscope," *IEEE Transaction on Biomedical Engineering*, vol. 44, no. 8, pp. 753-759.

[38] Tong K.Y. and Granat M.H., "A practical gait analysis system using gyroscopes," *Med. Eng.Phys.*, vol. 21, pp. 87-94, 1999.

[39] Mayagoitia R.E., Nene A.V., and Veltink P.H., "Accelerometer and rate gyroscope measurement of kinematics: an inexpensive alternative to optical motion analysis systems," *Journal of Biomechanics*, vol. 35, pp. 537-543, 2002.

[40] Pappas I.P., Popovic M.R., Keller T., Dietz V., and Morari M., "A reliable gait phase detection system," *IEEE Transaction on Neural System and Rehabilitation Engineering*, vol. 9, pp. 113-125, 2001.

[41] Bachmann E.B., *Inertial and Magnetic Tracking of Limb Segment Orientation for Inserting Humans into Synthetic Enviroment*.: Naval Postgraduate school, 2000, Ph.D. Dissertation.

[42] Sabatini A.M., "Quaternion-based strap-down integration method for applications of inertial sensing to gait analysis," *Medical & Biological Engineering & Computing*, vol. 43, pp. 94-101, 2005.

[43] Luinge H.J. and P.H. Veltink, "Measuring orientation of human body segments using miniature gyroscopes and accelerometers," *Medical & Biological Engineering & Computing*, vol. 43, pp. 273-282, 2005.

[44] Dejnabadi H., Jolles B.M., and K. Aminian, "A New Approach to Accurate Measurement of Uniaxial Joint Angles Based on a Combination of Accelerometers and Gyroscopes," *IEEE Transactions of Biomedical Engineering*, vol. 52, no. 8, pp. 1478-1485, 2005.

[45] Hyde R.A., Ketteringham L.P., Neild S.A., and Jone R.J.S., "Estimation of Upper-Limb Orientation Based on Accelerometer and Gyroscope Measurements," *IEEE Transactions on Biomedical Engineering*, vol. 55, no. 2, pp. 746-785, February 2008.

[46] Roetenberg D., *Inertial and magnetic sensing of human motion*.: University of Twente, 2006, Ph.D. Dissertation.

[47] Roetenberg D., Slycke P.J., and Veltink P.H., "Ambulatory Position and Orientation Tracking Fusing Magnetic and Inertial Sensing," *IEEE Transaction on Biomedical Engineering*, vol. 54, no. 5, pp. 883-891, May 2007.

[48] Zhu R. and Zhou Z., "A Real-Time Articulated Human Motion Tracking Using Tri-Axis Inertial/Magnetic Sensors Package," *IEEE Transaction on Neural Systems and Rehabilitation Engineering*, vol. 12, no. 2, pp. 295-303, June 2004.

[49] Kawano K. et al., "Analyzing 3D Knee Kinematics Using Accelerometers, Gyroscopes and Magnetometers," in *IEEE International Conference on System of Systems Engineering, 2007. SoSE '07.*, April, 2007, pp. 1-6.

[50] Foxlin E., Harrington M., and Pfeifer G., "Constellation: A wide range wireless motion tracking system for augmented reality and virtual set applications," in *Proceedings of SIGGRAPH*, Orlando, FL., 1998.

[51] Jonker P., Persa S., Caarls J., de. Jong F., and Lagendijk I., "Philosophies and Technologies for Ambient Aware Devices In Wearable Computing grid," *Computer Communications*, vol. 26, no. 11, pp. 1145-1158, 2003.

[52] Wisniowski H., "Analog Devices And Nintendo Collaboration Drives Video Game Innovation With iMEMS Motion Signal Processing Technology," Analog Device., Inc, 2006.

[53] Kärrholm J., "Roentgen stereophotogrammetry: Review of orthopedic applications," *Acta Orthopaedica*, vol. 60, no. 4, pp. 491-503, 1989.

[54] Selvik G., "Roentgen stereophotogrammetry A method for the study of the kinematics of the skeletal system," *Acta orthopaedica*, vol. 60, no. 232, pp. 1-51, 1989.

[55] Kalman R.E., "A New Approach to Linear Filtering and Prediction Problems," *Transaction of the ASME - Journal of Basic Engineering*, vol. 82, pp. 35-45, 1960.

[56] Smith G.L., Schmidt S.F., and McGee L.A., "Application of Statistical Filter Theory to the Optimal Estimation of Position and Velocity on Board a Circumlunar Vehicle," NASA, 1962.

[57] Wahba G., "A Least-Squares Estimate of Satelite Attitude," *SIAM Review*, vol. 7, no. 3, p. 409, 1965.

[58] Lerner G.M., "q Method," in *Spacecraft Attitude Determination and Control*, Wertz J.R., Ed.: Kluwer Academic Publishers, 1978, pp. 426-428.

[59] Shuster M.D. and S.D. Oh, "Three-Axis Attitude Determination from Vector Observations," *Journal of Guidance and Control*, vol. 4, no. 1, pp. 70-77, 1981.

[60] Shuster MD, "Kalman Filtering of Spacecraft Attitude and Quest Model," *The journal of Astronautical Sciences*, vol. 38, no. 3, pp. 377-393, July_September 1990.

[61] Bar-Itzhack IY, "REQUEST: A Recursive QUEST Algorithm for Sequential Attitude Determination," *Journal of Guidance, Control, and Dynamics*, vol. 19, no. 5, pp. 1034-1038, September-October 1996.

[62] Choukroun D, Bar-Itzhack IY, and Y. Oshman, "Optimal REQUEST Algorithm for Attitude Determination," *Journal of Guidance, Control, and Dynamics*, vol. 27, no. 3, pp. 418-425, May-June 2004.

[63] M.L. Psiaki, "Attitude-Determination Filtering via Extended Quaternion Estimation,"

Journal of Guidance, Control, and Dynamics, vol. 23, no. 2, pp. 206-214, 2000.

[64] Julier S.J. and Uhlmann J.K., "A General Method for Approximating Nonlinear Transformations of Probability Distributions (Technical Report)," Robotics Research Group; Department of Engineering Science; University of Oxford., Oxford, UK, 1996.

[65] Gordon N.J., D.J. Salmond, and A.F.M. Smith, "Novel Approach to Nonlinear/Non-Gaussian Bayesian State Estimation," *IEE Proceedings-F*, vol. 140, pp. 107-113, 1993.

[66] Metropolis N. and Stanislaw U., "The Monte Carlo Method," *Journal of American Statistical Association*, vol. 44, pp. 334-341, 1949.

[67] A Doucet, S Godsill, and C Andrieu, "On Sequential Monte Carlo Sampling Methods for Bayesian Filtering," *Statistics and Computing* , vol. 10, pp. 197-208, 2000.

[68] Bergman N., "Recursive Bayesian estimation: Navigation and tracking applications," Linköping University, Linköping, Sweden, Linköping, Sweden, Ph.D. Dissertation 1999.

[69] Liu JS and Chen R, "Sequential Monte Carlo methods for dynamics systems," *Journal of the American Statistical Association*, vol. 93, no. 443, pp. 1032-1044, 1998.

[70] Osder S.S., Rouse W.E., and Young L.S., "Navigation, Guidance and Control Systems for V/STOL aircraft," *Sperrt Tech.*, vol. 1, no. 3, 1973.

[71] W.T. Higgins, "A Comparsion of Complementary and Kalman Filtering," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 11, no. 3, pp. 321-326, May 1975.

[72] R Mahony, "Nonlinear Complementary Filters on the Special Orthogonal Group," *IEEE Transaction on Automatic Control*, vol. 53, no. 3, pp. 1203-1218, June 2008.

[73] Glister G., "High accuracy performance capabilities of the military standard ring laser gyro inertial navigation unit," in *IEEE Position Location and Navigation Symposium*, April, 1994.

[74] Honeywell. GG1320AN digital laser gyro specification. [Online]. http://www51.honeywell.com/aero/common/documents/myaerospacecatalog-documents/Missiles-Munitions/GG1320AN_Digital_Laser_Gyro.pdf

[75] Schmidt S.F. McGee L.A., "Discovery of the Kalman Filter as a Practical Tool for Aerospace and Industry," *NASA Technical Memorandum 86847*, November 1985.

[76] P.C. Hughes, *Spacecraft Attitute Dynamics*. New York: John Wiley & Sons, 1986.

[77] Lefferts EG, Markley FL, and Shuster MD, "Kalman Filtering for Spacecraft Attitude Estimation," *Journal of Guidance, Control and Dynamics*, vol. 5, no. 5, pp. 417-429, September-October 1982.

[78] Bar-Itzhack IY and Oshman Y., "Attitude Determination from Vector Observations: Quaternion Estimation," *IEEE Transaction on Aerospace and Electronic System*, vol. AES-21, no. 1, pp. 128-136, Janurary 1985.

[79] Kasdin NJ, "Satellite Quaternion Estimation from Vector Measurements with the Two-step Optimal Estimator," in *AAS Guidance and Control Conference*, February 2002, pp. AAS 02-002.

[80] Goddard J.S. and Abidi M.A., "Pose and Motion Estimation Using Dual Quaternion-Based Extended Kalman Filtering," Oak Ridge National Laboratory, Oak Ridge, TN, ORNL/CP--96358, 1998.

[81] Marins J.L., Yun X., Bachmann E.R., McGhee R.B., and Zyda M.J., "An Extended Kalman Filter for Quaternion-Based Orientation Estimation using MARG Sensors," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2001, pp. 2003-2011, vol. 4.

[82] Crassidis J.L. and Markley F.L., "Unscented Filtering for Spacecraft Attitude Estimation," *Journal of Guidance, Control and Dynamics*, vol. 26, no. 4, pp. 536-542, 2003.

[83] Harada T., Uchino H., Mori T., and Sato T., "Portable Orientation Estimation Device Based on Accelerometers, Magnetometers and Gyroscope Sensors for Sensor Network," in *IEEE Conference on Multisensor Fusion and Integration for Intelligent System*, 2003, pp. 191-197.

[84] Madgwick S.O.H., "Estimation of IMU and MARG orientation using a gradient descent algorithm," in *IEEE International Conference on Rehabilitation Robotics (ICORR)*, June 2011, pp. 1-7.

[85] Yang Y. and Li J., "Particle Filtering for Gyroless Attitude/Angular Rate Estimation Algorithm," in *2010 First International Conference on Pervasive Computing, Signal Processing and Applications*, 2012, pp. 1188-1191.

[86] Cheng Y and JL Crassidis, "Particle Filtering for Sequential Spacecraft Attitude Estimation," in *Proceedings of AIAA Guidance Navigation and Control Conference*, Rhode Island, Reston, VA, August 2004, pp. AIAA 2004-5337.

[87] Yang S., Wen D., Sun J., and Ma J., "Gaussian Sum Particle Filter for Spacecraft Attitude Estimation," in *2010 2nd International Conference on Signal Processing Systems (ICSPS)*, 2010, pp. 566-570.

[88] Carmi A, *Sequential Monte Carlo Methods for Spacecraft Attitude and Angular Rate Estimation from Vector Observations*.: Israel Institute of Technology, 2008, Research Thesis.

[89] Mackenzie J.I., *The topology of Stiefel Manifolds*. London , UK: Cambridge University Press , 1976.

[90] Kanti M. and Jupp P.E., *Directional Statistics*. London, UK: John Wiley & Sons, Ltd, 1999.

[91] Downs TD, "Orientation statistics," *Biometrika*, vol. 59, pp. 665-676, 1972.

[92] Khatri C.G. and Mardia K.V., "The Von Mises-Fisher Matrix Distribution in Orientation Statistics," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, no. 1, pp. 95-106, 1977.

[93] Piater J.H., *Visual Feature Learning*, Ph.D. thesis ed. Meassachussets, USA: University of Massachussets, 2001.

[94] Mooney J.A., Helms P.J., and Jolliffe I.T., "Fitting mixtures of von Mises distributions: a case study involving sudden infant death syndrome," *Computational Statistics & Data Analysis*, vol. 41, pp. 505-513, 2003.

[95] Wallace C.S. and Dowe D.L., "MML Clustering of Multi-State, Poisson, von Mises circular and Gaussian distribution," *Statistics and Computing* , vol. 10, no. 1, pp. 73-83, January 2000.

[96] Wood A.T.A., "Simulation of the von Mises Distribution," *Communications of Statistics, Simulation and Computation*, vol. 23, pp. 157-164, 1994.

[97] Ulrich G., "Computer generation of distributions on the m-spheres," *Applied Statistics* , vol. 33, pp. 153-163, 1984.

[98] Hoff P.D., Gibbs Sampling of the Bingham-von Mises-Fisher distribution, with an application to protein interaction networks, 2007.

[99] Glover J., Rusu R., and G., Bradski , "Monte Carlo Pose Estimation with Quaternion Kernels and the Bingham Distribution," in *Proceedings of Robotics: Science and Systems*, Los Angeles, CA, 2011.

[100] Markley F.K., Cheng Y., Crassidis J.L., and Oshman Y., "Quaternion Averaging,".

[101] Buss S.R. and J.P. Fillmore, "Spherical Averages and Applications to Spherical Splines and Interpolation," *ACM Trans. Graphics*, vol. 20, no. 2, pp. 95-126, 2007.

[102] Clark R.M. and Thompson R., "Statistical comparsion of palaeomagnetic directional records from lake sediment," *Geophys. J. R. Astr. Soc*, vol. 76, pp. 337-368, 1984.

[103] Shoemake K, "Animating rotation with quaternion curves," *ACM SIGGRAPH*, pp. 245-254, 1985.

[104] Kitagawa G., "Monte Carlo Filter and Smoother for Non-Gaussian Nonlinear State Space Models," *Journal of Computational and Graphical Statistics*, vol. 5, no. 1, pp. 1-25, 1996.

[105] Pitt MK and Shephard N, "Filtering via Simulation: Auxiliary Particle Filters," *Journal of the American Statistical Association*, vol. 94, no. 446, pp. 590-599, June 1999.

[106] NINDS, "Low Back Pain Fact Sheet," NIH, NIH publication No: 03-5161, 2003.

[107] Komistek R.D., Mahfouz M.R., and Cheng J.S., "Innvoative Tools for In Vivo Computation Prediction of Lumbar Stresses," NIH, 2012.

APPENDIX

A. Appendix A

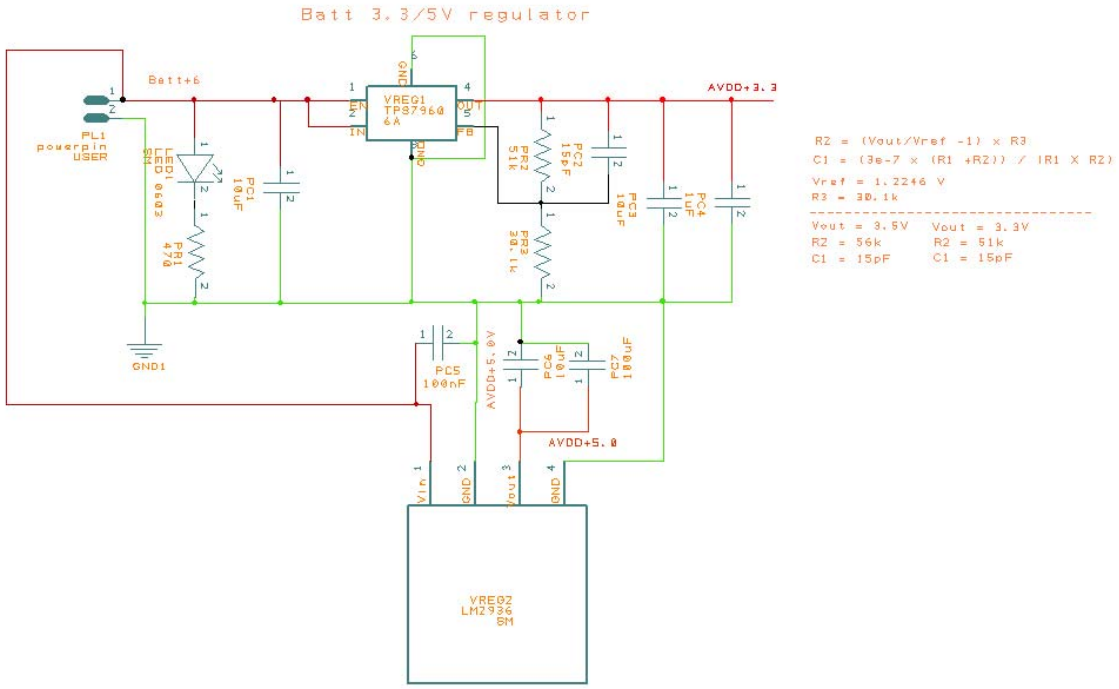

Figure A-1 - Schematic for powering circuit

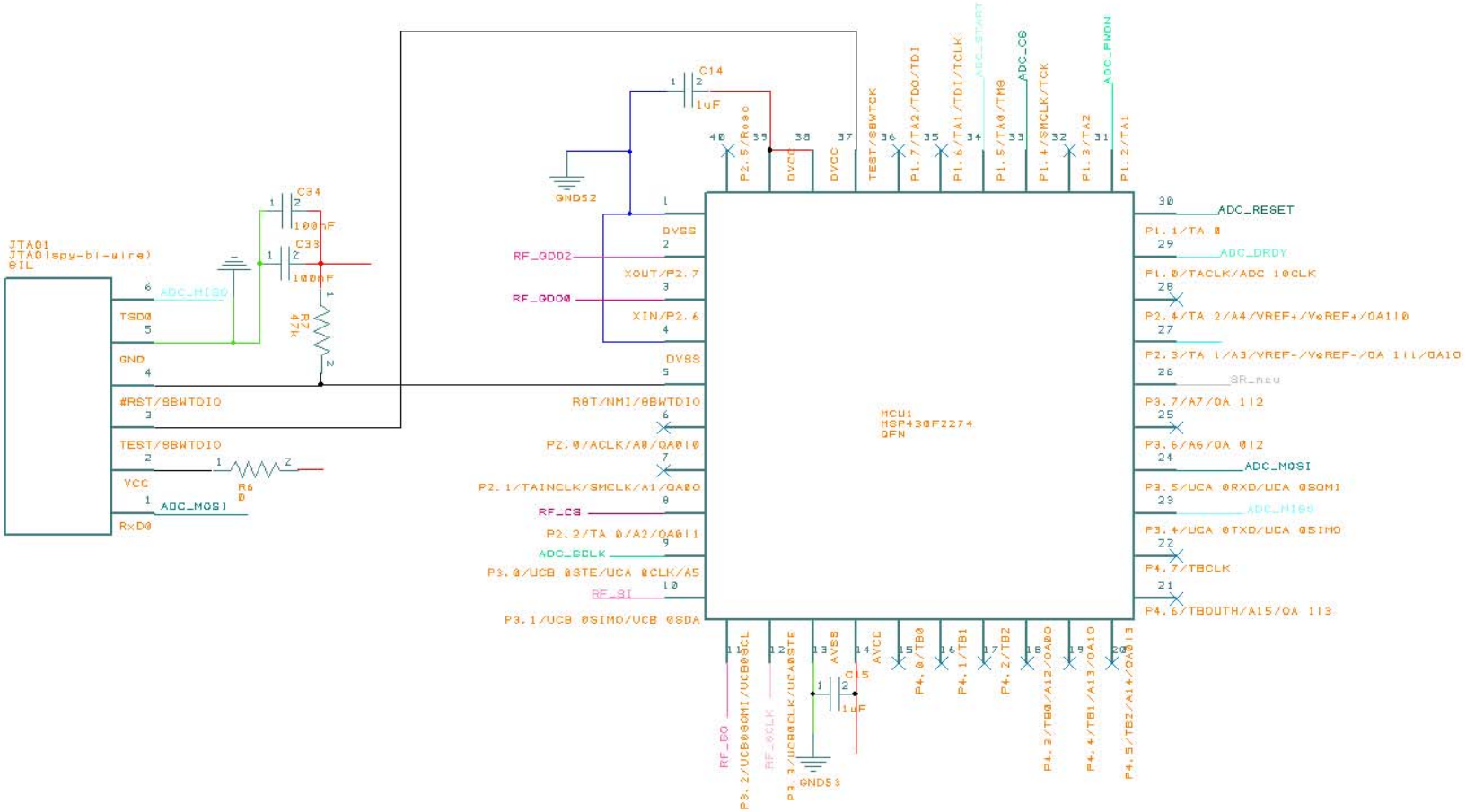

Figure A-2: Schematic for Microcontroller and JTAG setup

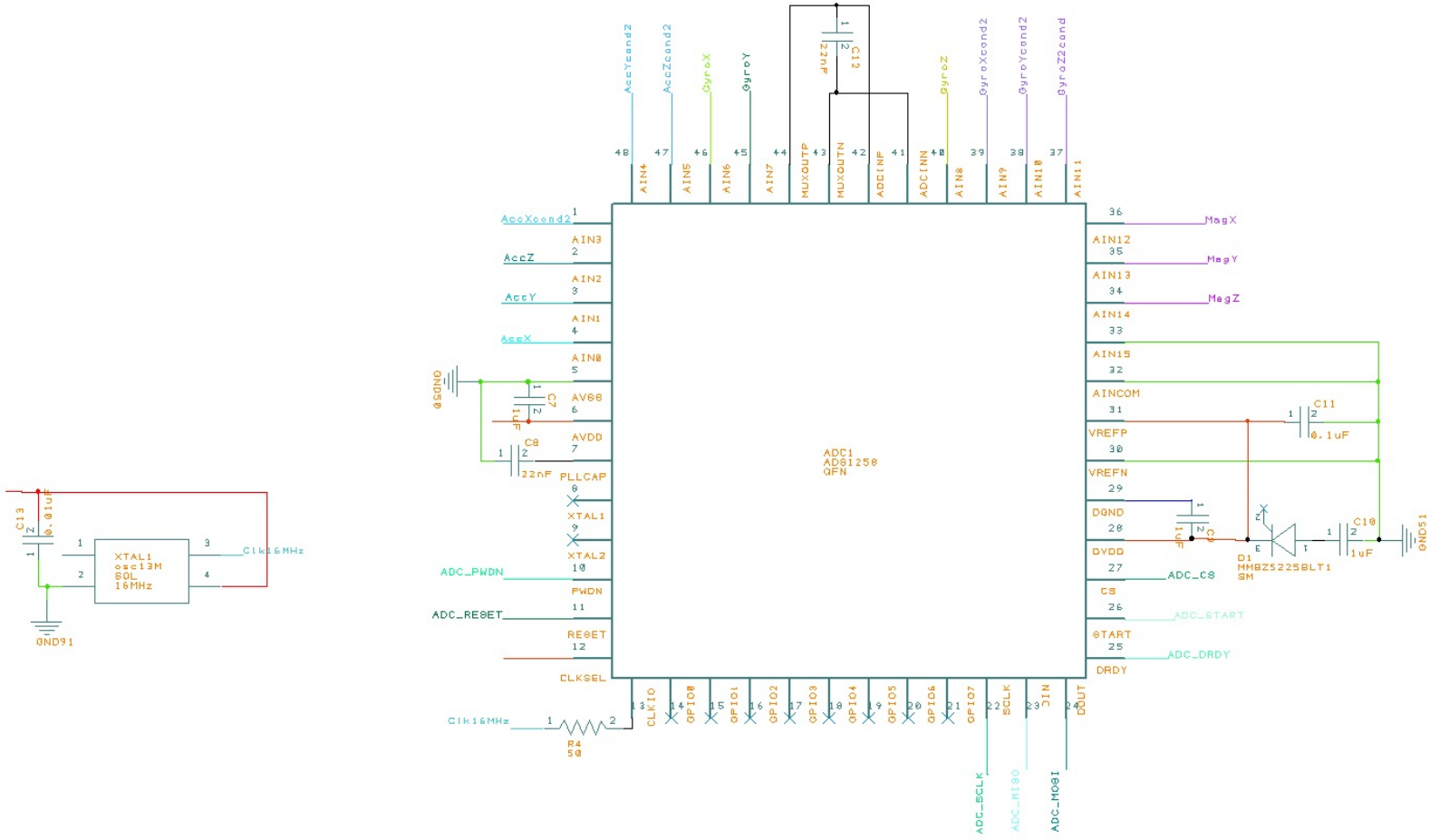

Figure A-3: Schematic for ADC and clock

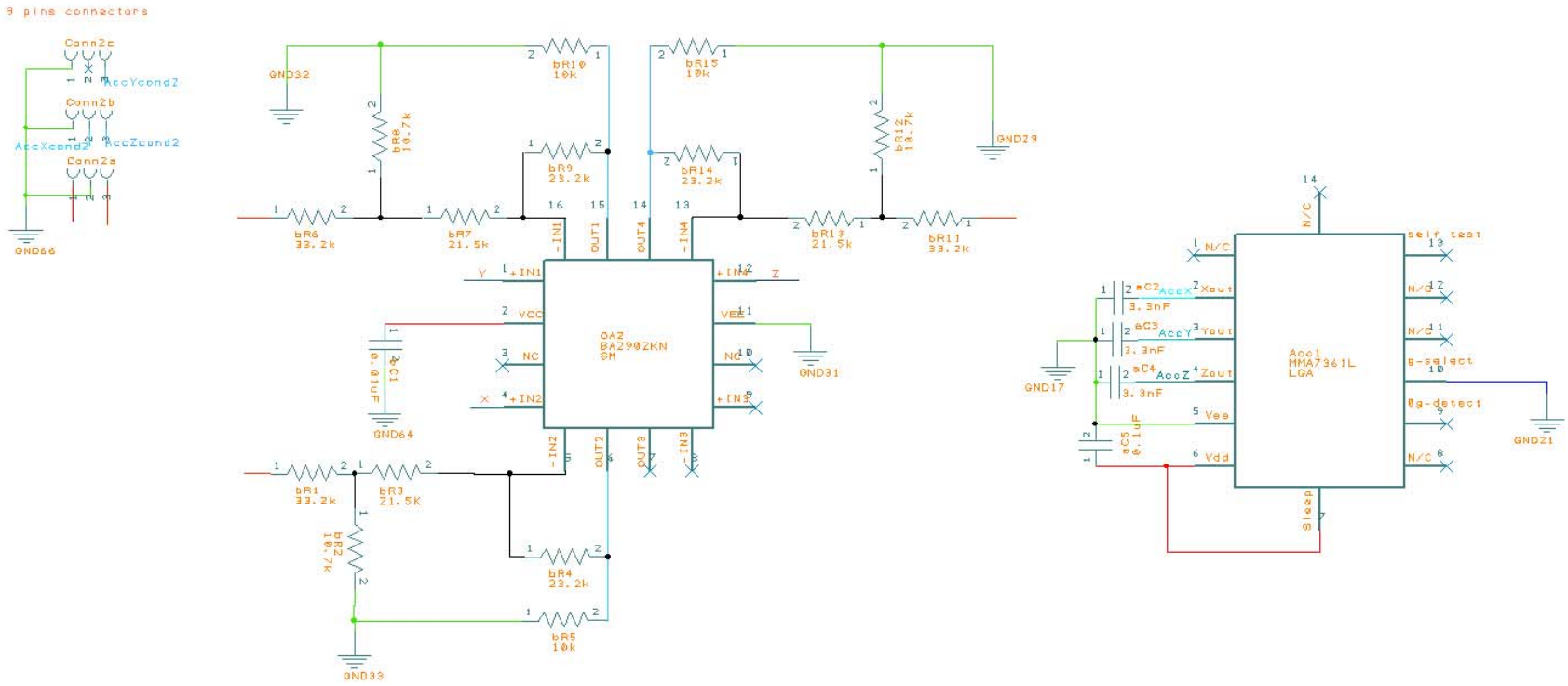

Figure A-4: Schematic for Accelerometers and its signal conditioning circuit (1.5g setup)

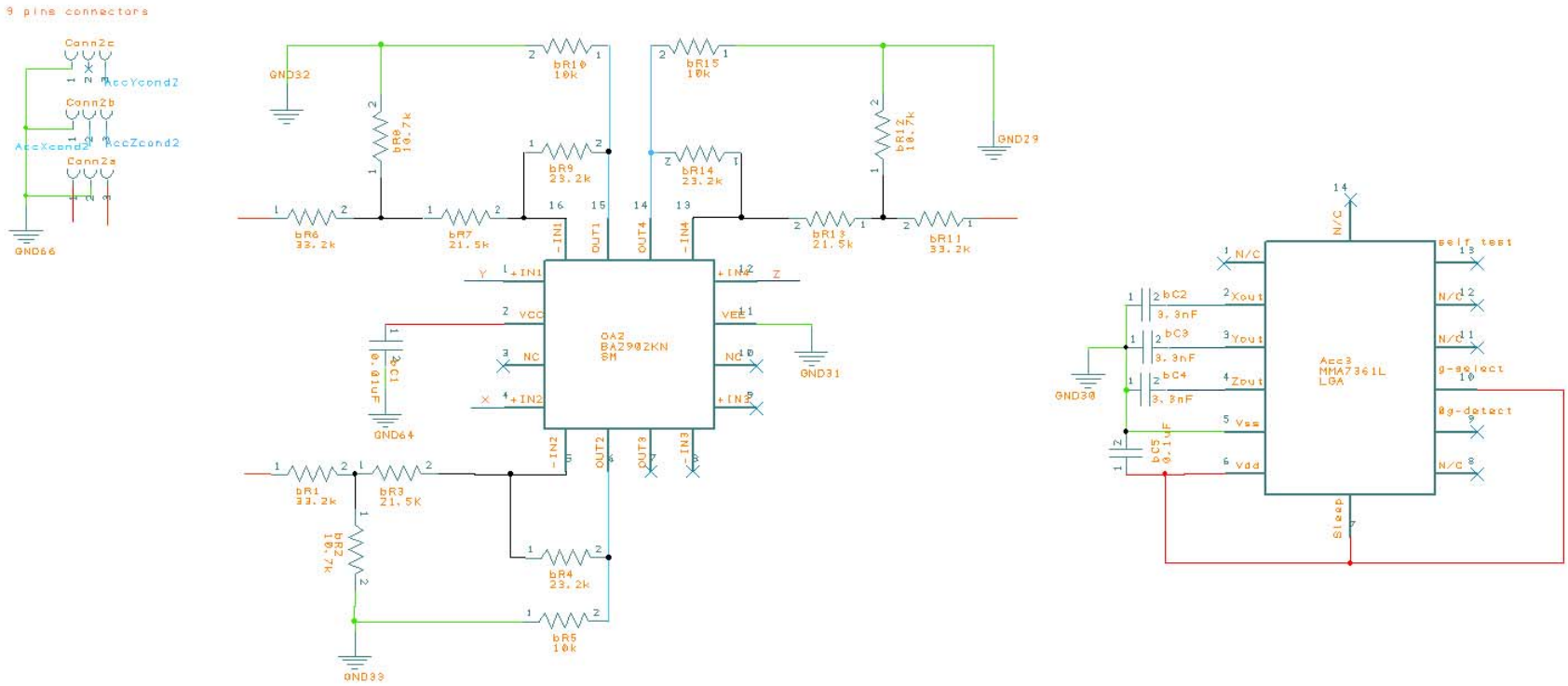

Figure A-5: Schematic for Accelerometer and its signal conditioning circuit (6g setup)

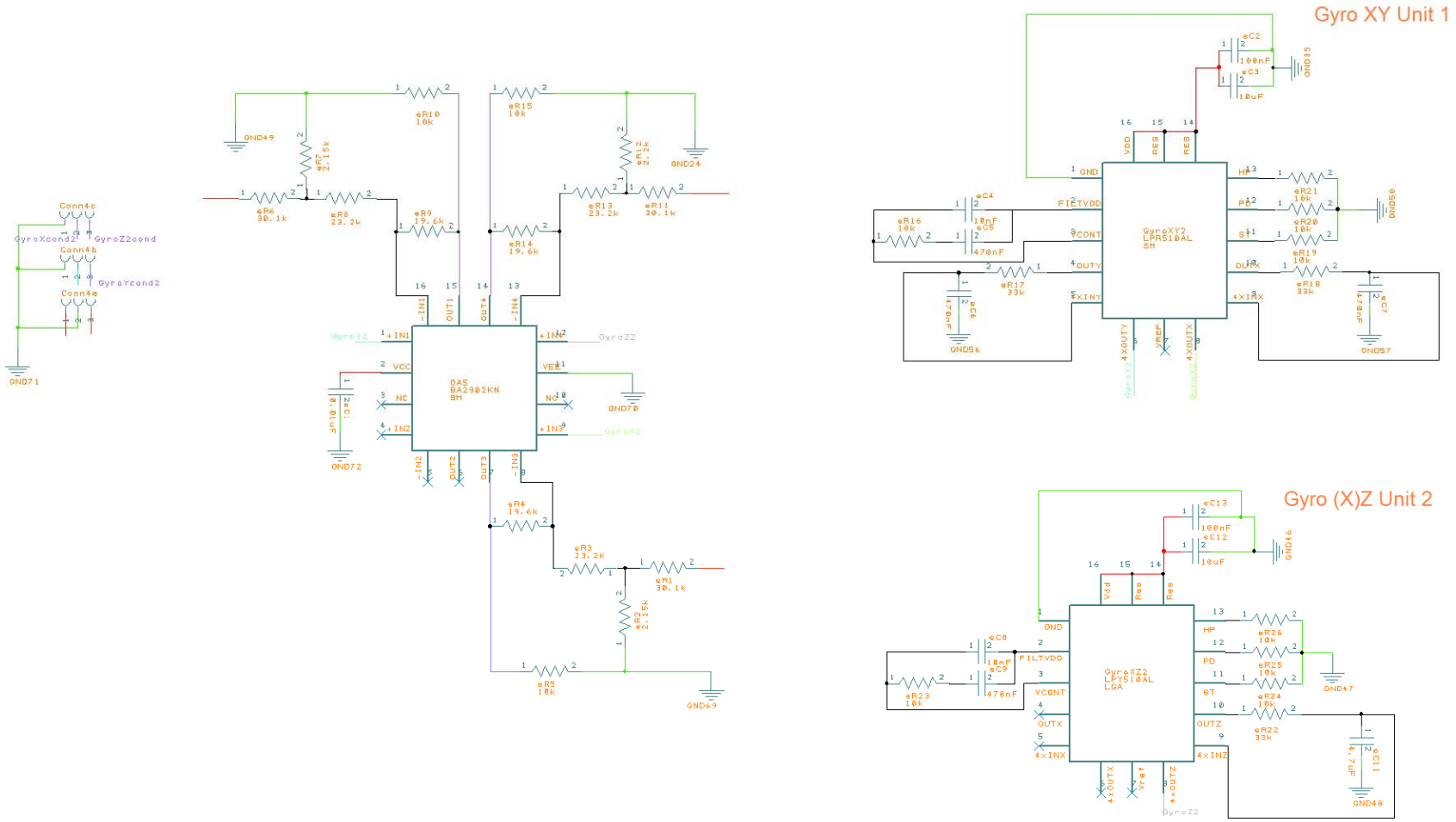

Figure A-6: Schematic for Gyroscopes and its signal conditioning circuit (+/- 100 dps setup)

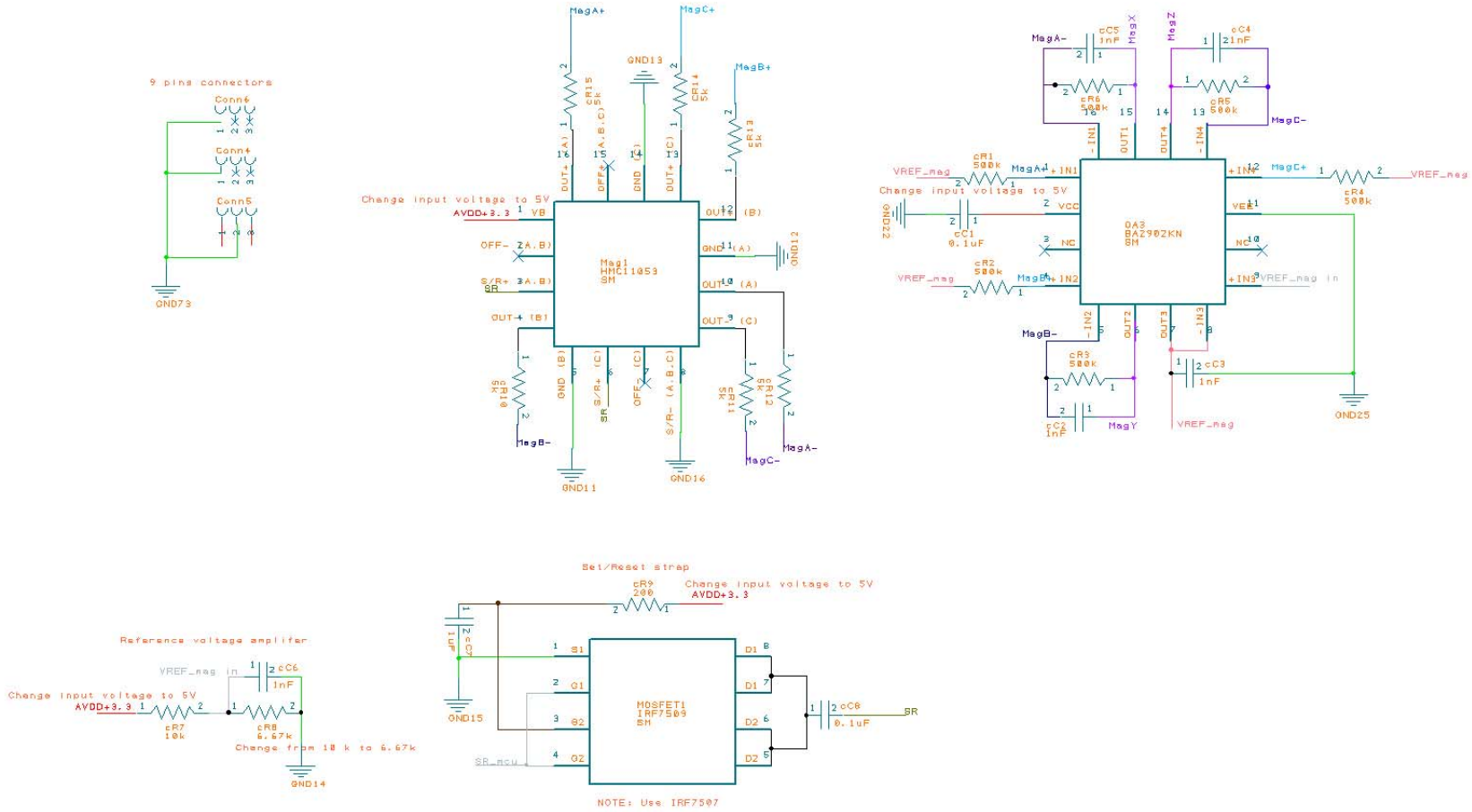

Figure A-7: Schematic for Magnetometer its signal conditioning circuit and set/reset trigger

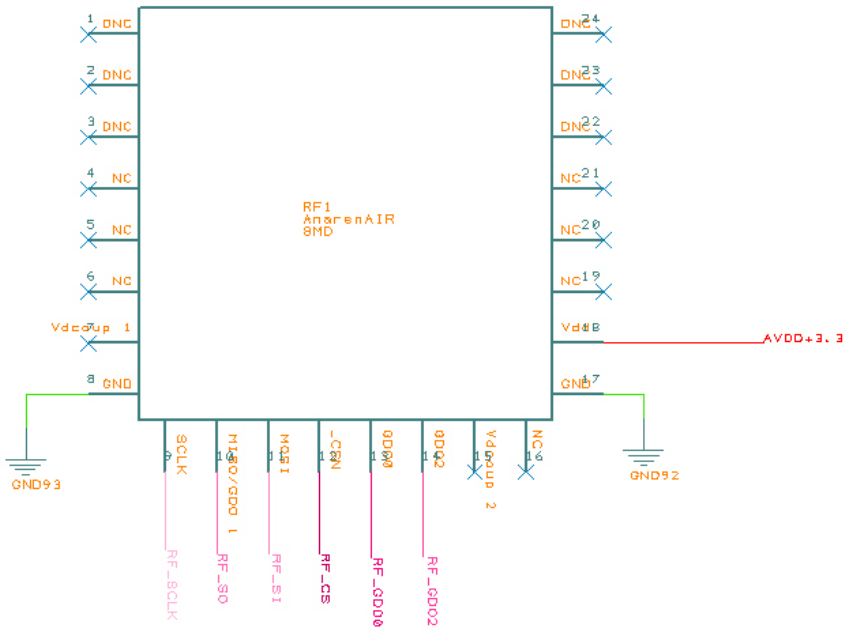

Figure A-8: Schematic for wireless transmitter interface

B. Appendix B

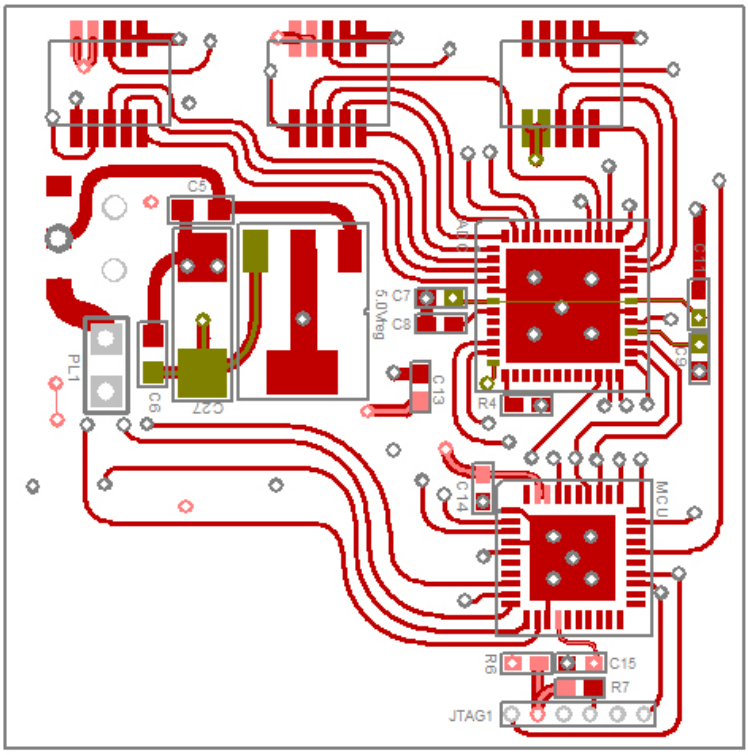

Figure B-1: Layout for signal processing board (Top Copper)

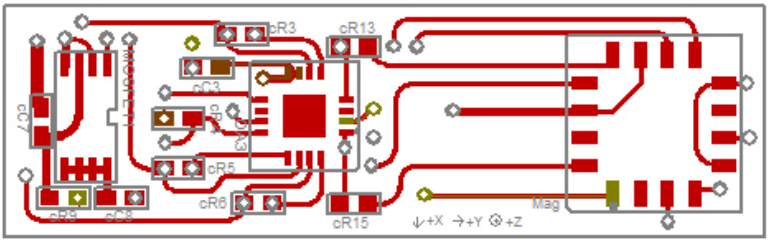

Figure B-2: Layout for magnetometer sensor strip (Top Copper)

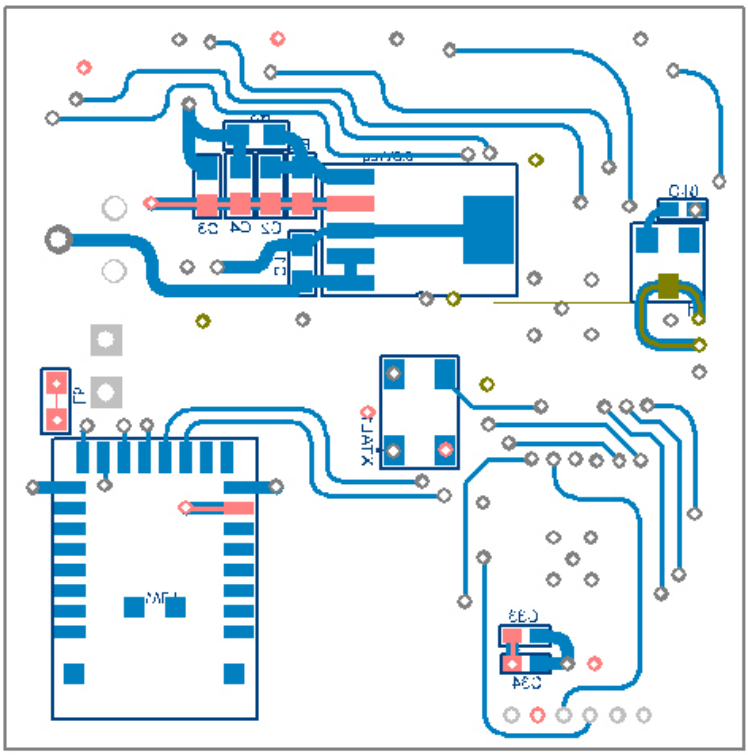

Figure B-3: Layout for signal processing board (Bottom Copper)

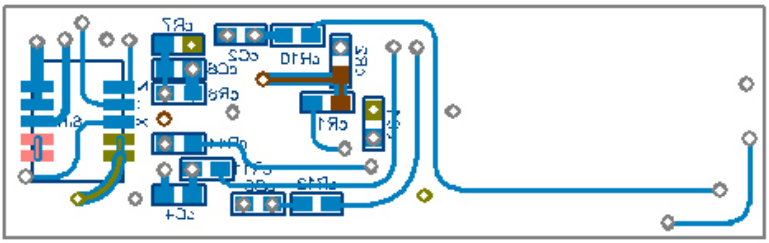

Figure B-4: Layout for magnetometer sensor strip (Bottom Copper)

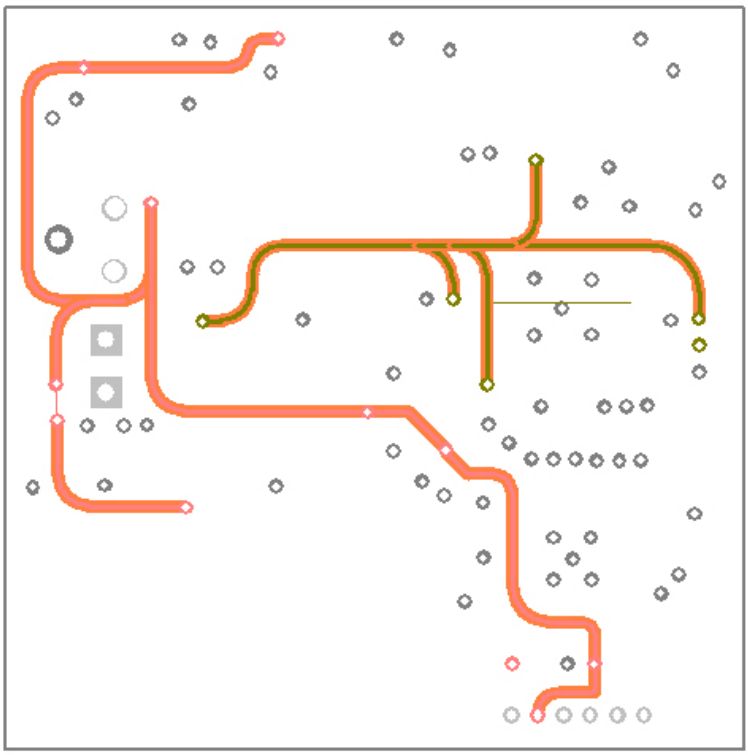

Figure B-5: Layout for signal processing board (Inner Layer: Power)

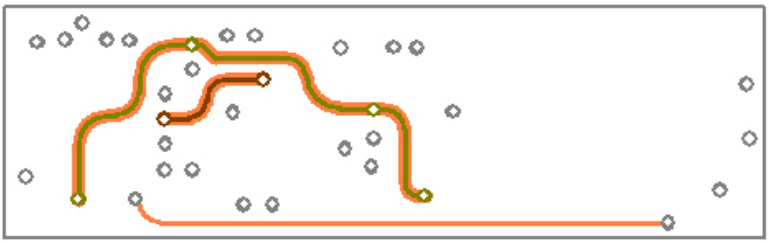

Figure B-6: Layout for magnetometer sensor strip ((Inner Layer: Power)

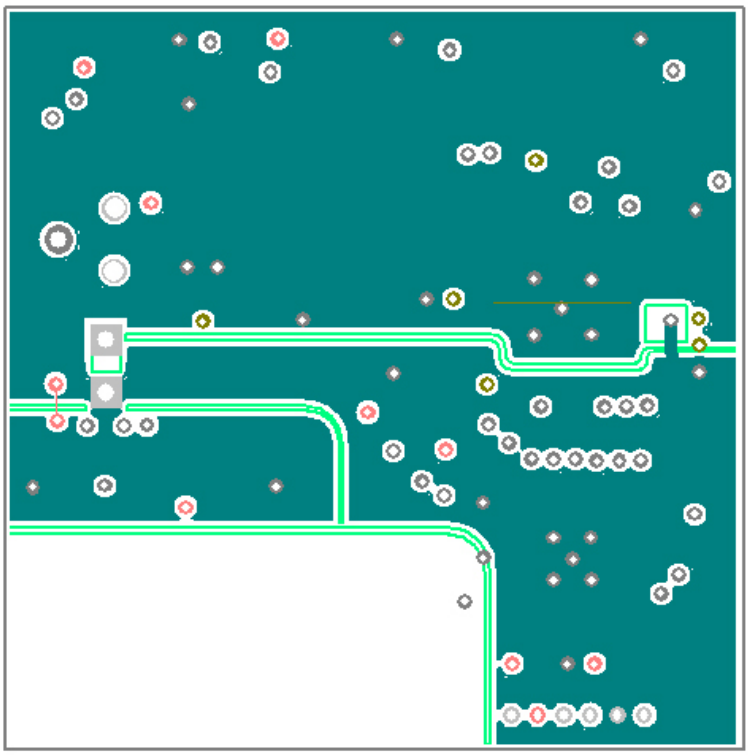

Figure B-7: Layout for signal processing board (Inner Layer: Ground)

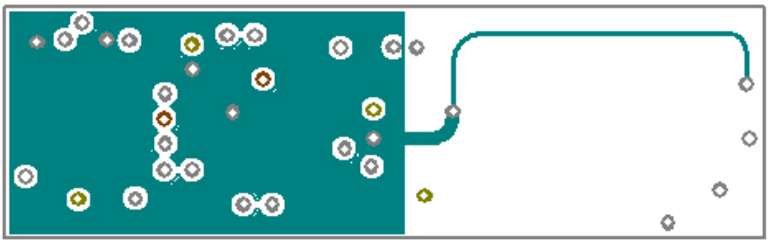

Figure B-8: Layout for magnetometer sensor strip ((Inner Layer: Ground)

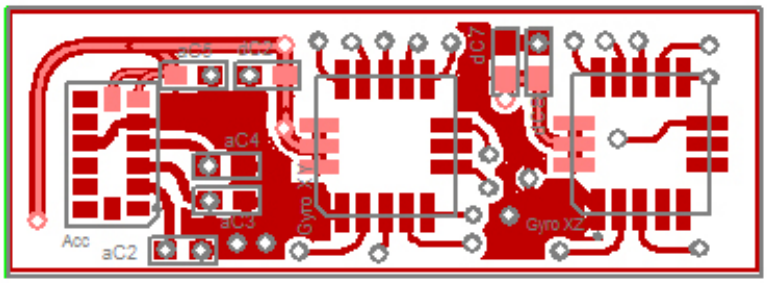

Figure B-9: Layout for accelerometers and gyroscopes sensor strip (Top Copper)

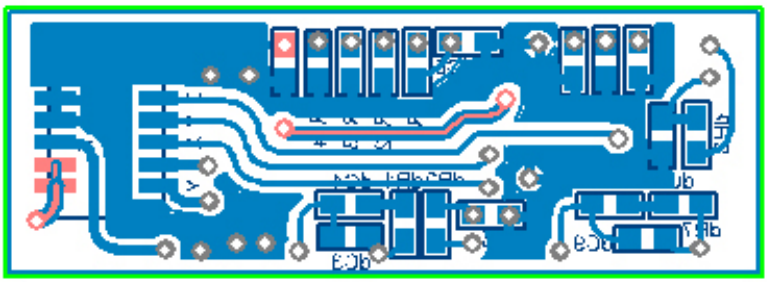

Figure B-10: Layout for accelerometers and gyroscopes sensor strip (Bottom Copper)

C. Appendix C

Using the statistical simulation methods discussed in chapter 6.5, an analysis was conducted to examine the quaternion to modified Rodriguez parameters (MRP) convention. In the analysis, vMF and NU densities were tested. Quaternions were generated from both of the densities and were converted to MRP. The expectation was calculated in MPR and converted back into quaternion. The expectation of the original quaternion is then compared with the MPR converted quaternion. In general, the difference between the two expectations is trivial regardless of the densities or offset rotations as shown from the two rotations sets in Figure C-1.However, if offset rotation approaches 2π, there is a discrepancy between the expectations between the original and MRP converted quaternions. The rotation set in Figure C-2 and C-3 show the expectations differences using quaternions sampled from vMF and NU densities respectively.

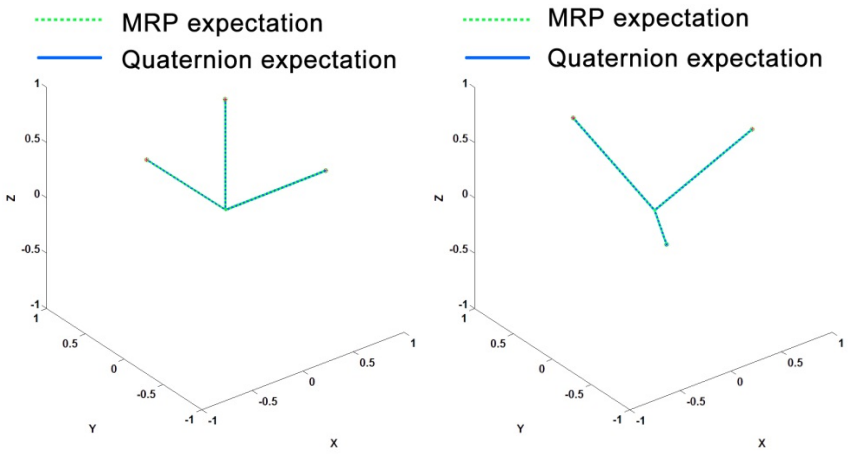

Figure C-1: Expectations between quaternion and MRP converted quaternion; (Left) no offsets, (Right) offset rotation [45 45 45]

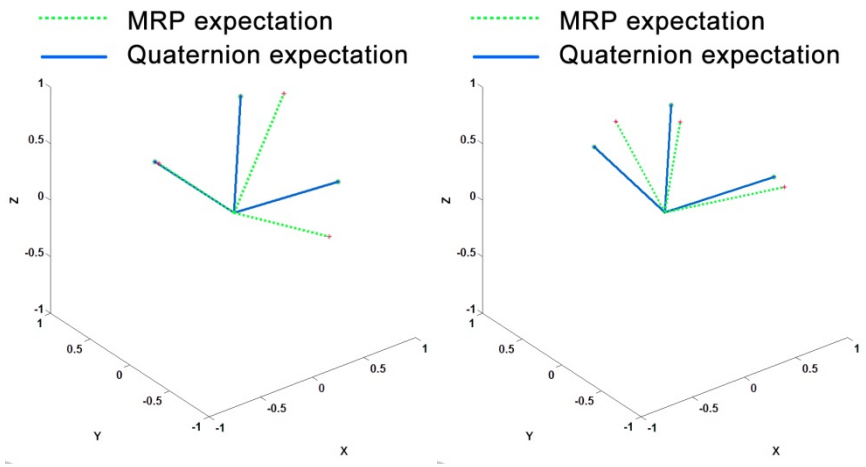

Figure C-2: Comparsion between the expectations of the original quaternion sampled from vMF density and the quaternion converted from the MRP expectation; (Left) offset rotation: [0 4 359], (Right) offset rotation: [355 0 5]

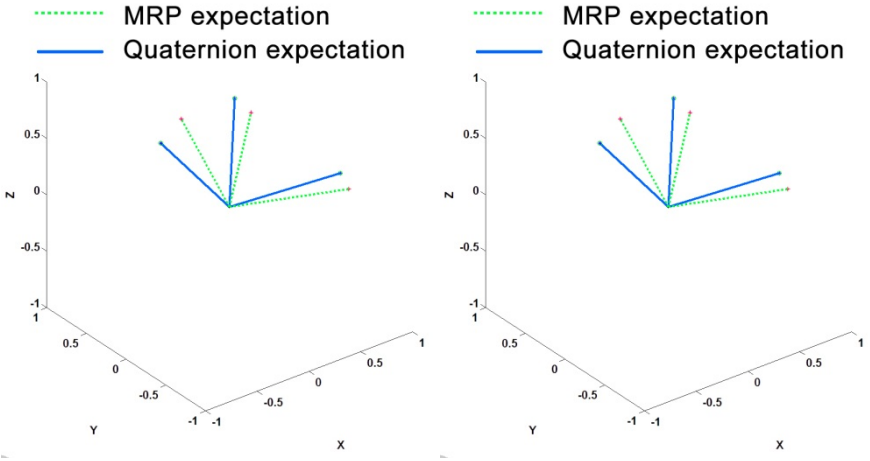

Figure C-3: Comparsion between the expectations of the original quaternion sampled from NU density and the quaternion converted from the MRP expectation; (Left) offset rotation: [0 4 359], (Right) offset rotation: [355 0 5]

D. Appendix D

Table D-1- Pseudo code of Deterministic Resampling Technique [104]

Input: $q(i)$ (Particles),
$w(i)$ (Normalized importance weight),
N (number of particles)

1. $k_j := [0]_{1\to N}$
2. Determine the cumulative distribution of $q(i)$, $(p(q))$
3. $a \sim \Pi[0,1]$
4. $S = (a \to (N - 1 + a))/N$
5. Define $j = 1$
6. For $i: 1 \to N$
7. While $S > p(q)$
8. $j++$
   End
9. k_j++
   End
10. Define index $n = 1$
11. For $i: 1 \to N$
12. If $k_j > 1$
13. For $j: n \to n + k_j(i) - 1$
14. $n_{resample}(j) = n(i)$
    End
    End
15. $n = n + k_j(i)$
    End
    Return $n_{resample}$

Table D-2- Pseudo code of Residual Resampling Technique [69]

Input: $q(i)$ (Particles),
$w(i)$ (Normalized importance weight),
N (number of particles)

1. $k_j := [0]_{1\to N}$
2. $q_{residual}(i) = N(q(i)^T)$
3. $k_j = truncate(q_{residual}(i))$
4. $k_{residual} = N - \sum k_j$
5. If $k_{residual} \cong 0$
6. $\quad q_{residual}(i) = \frac{(q_{residual}(i) - k_j)}{k_{residual}}$
7. $\quad$ Determine the cumulative distribution of $q_{residual}(i)$, $(p(q_{residual}))$
8. $\quad$ Generate $k_{residual}$ ordered random variables (U) distributed between 0 and 1
9. $\quad$ Define $j = 1$
10. $\quad$ For $i: 1 \to k_{residual}$
11. $\quad\quad$ While $U > p(q_{residual})$
12. $\quad\quad\quad j++$
$\quad\quad$ End
13. $\quad\quad k_j++$
$\quad$ End
End
14. Define index $n = 1$
15. For $i: 1 \to N$
16. $\quad$ If $k_j > 1$
17. $\quad\quad$ For $j: n \to n + N(i) - 1$
18. $\quad\quad\quad n_{resample}(j) = n(i)$
$\quad\quad$ End
$\quad$ End
19. $\quad n = n + N(i)$
End
Return $n_{resample}$

Table D-3- Pseudo code of Auxiliary Resampling Technique [105]

Input: $q(i)$ (Particles),
$w(i)$ (Normalized importance weight),
N (Number of particles)
α (Regularize constant)

1. $k_{auxiliary} = \frac{(\alpha-1)q(i)+\overline{q(i)}}{\alpha}$
2. $k_{auxiliary} = k_{auxiliary} / \sum q(i)$
3. For $i: 1 \rightarrow N$
4. $U \sim \Pi[0,1]$
5. $k_j = 0$
6. For $j: 1 \rightarrow N$
7. $k_j = k_j + k_{auxiliary}$
   End
8. If $k_j \geq U$
9. $q_r(j) = q(i)$
   End
   End
   Return q_r

Vita

Gary To was born in Hong Kong in 1982. He began his university study at the University of Tennessee, Knoxville in 2000 and received the B.S. degree in biomedical engineering with a minor in material science and engineering in 2004. He was awarded as the outstanding engineering senior by the Engineering Honor society Tau Beta Pi. He continued his study in the master program in biomedical engineering under the guidance of Dr. Mohamed Mahfouz. His master research involved using piezo-resistive microcantilevers for intra-operative strain sensing device for total knee replacement surgery. He worked closely with embedded system and electronics designs. He earned his M.S. degree in 2007. He received the citation for extraordinary professional promise from the University of Tennessee in 2006.

He has published and presented in many international conference in the fields of sensing technologies in biomedicine. He is an active member within the IEEE. He is also active on many professional and hobby online electronics forums, including edaboard.com, the TI forum and sparkfun forum, and provides advice in circuit and system designs. His current research interests include embedded system design, tracking system using Simultaneous Localization And Mapping for medical applications, and Bayesian algorithm design with FPGA. He has a hobby of building autonomous robots.