

# Lower Limb Motion Estimation

---

Kinematic Modelling and Estimation of the Gait using Cameras

and an IMU



Prepared by:

**Johann Hendrik Joosten**

Department of Electrical Engineering  
University of Cape Town

Prepared for:

**Dr. Amir Patel**

Department of Electrical Engineering  
University of Cape Town

Submitted to the Department of Electrical Engineering at the University of Cape Town  
in partial fulfilment of the academic requirements for a Bachelor of Science degree in  
Mechatronics.

**November 12, 2017**

THIS PAGE INTENTIONALLY LEFT BLANK

# Terms of Reference

---

## Title

Lower Limb Motion Estimation - Kinematic Modelling and Estimation of the Gait using Cameras and an IMU

## Description

Recent breakthroughs in the field of artificial intelligence has invigorated the pursuit of humanoid robots. Unfortunately, modern bipedal robots lack the elegance of motion and fluidity observed in nature. Perhaps then a modern take on the lower limb kinematics of humans could provide insight to the field of bio-inspired robotics. By using modern cameras with minimal volume and accurate sensors, data capture systems can be transferred onto the subjects in question. This methodology allows for a much larger spectrum of motion capture and can greatly improve our understanding of movement in the unconstrained real world.

## Deliverables

The following items have been identified as critical deliverables for the project: Functional harness to hold data capture equipment. Kinematic model of the human lower-limbs. Estimation and fusion algorithm to process captured data.

## Skills and Requirements

Mechanical Design, Electrical Design, Programming and Modelling.

## Area

Computer Vision, Sensors, Biomechanics and Bio-inspired Robotics.

## Declaration

---

1. I know that plagiarism is wrong. Plagiarism is to use another's work and pretend that it is one's own.
2. I have used the IEEE convention for citation and referencing. Each contribution to, and quotation in, this report from the work(s) of other people has been attributed, and has been cited and referenced.
3. This report is my own work.
4. I have not allowed, and will not allow, anyone to copy my work with the intention of passing it off as their own work or part thereof.



Signature:

Hendrik Joosten

November 12, 2017

## Acknowledgements

---

Firstly, I would like to thank Dr. Amir Patel for supervising this project, without whom this work would not have been possible. Had it not been for his incredible insight and creative thinking the device designed may never have existed. Also, to Neil Steenkamp for helping me whenever I had questions and explaining difficult concepts in a simple and clear manner. Lastly, I would like to thank Bradley Stocks for the foundational work he has done in this field and his efforts in documenting it. His commitment and work ethic has helped and inspired me to take on such a complex topic.

I would like to acknowledge my peers who have helped me through one of the most challenging endeavours of my life: obtaining an engineering degree. There have been numerous times where success would not have been possible without their help and support. It has been a time of discovery for us, not only academically but in all aspects of life.

To the University of Cape Town and the Department of Electrical Engineering for the world class education provided to me by excellent faculty. Specifically to the Mechatronics Research Lab for developing its namesake program of study and providing an interesting and future orientated curriculum.

I would like to thank my friends who have shown me the importance of perseverance and the value of courage. To Zach, Jaydon, Patrick, Dan, Conor, Dean, Mike, Alvin, James, Imran, and Luke thank you for the most amazing 4 years.

Finally, my family who have supported and guided me through all of my life. My mother for her care and compassion, my father for his belief and trust, and my sister for her inspiration. Without your vision and faith I would not have been where I am today.

## Abstract

---

This research aims to extend work completed by the Mechatronics Research Lab at the University of Cape Town that studied the use of subject borne cameras to analyse, model and estimate the kinematic motion of a cheetah tail. Image processing has progressed significantly within the last few years and studying motion with subject borne cameras allows data capture in new unconstrained environments. Improvements in sensor technology and long standing techniques of data fusion and state estimation allow new models for complex motion to be created.

A methodology for understanding the bipedal motion of humans in natural environments using wearable data capture systems consisting of cameras and sensors is proposed. The system is designed using common engineering methods and a prototype constructed using widely available technologies. Cameras as sensors to capture motion data have been used on animals in past studies, but at the time of writing no wearable camera based system for human motion capture could be found.

This thesis documents the definition, design and construction of such a system, noting important hardware considerations, modularity, and modelling assumptions.

Testing is performed and a dataset successfully obtained. This dataset is processed and applied to a model by using an Extended Kalman Filter to better quantify critical elements of the human gait. The final results are analysed and discussed, followed by conclusions outlining strengths and drawbacks of the system.

The proposed system is novel and serves as a proof of concept that can be adapted and improved based on the availability of equipment and end user needs. The research is finalized by presenting possible avenues of future work to improve and simplify such a system.

# Contents

Terms of Reference . . . . .	3
Declaration . . . . .	4
Acknowledgements . . . . .	5
Abstract . . . . .	6
Glossary . . . . .	16
Symbols . . . . .	17
<b>1 Introduction</b>	<b>18</b>
1.1 Background to the Study . . . . .	18
1.2 Objectives of the Study . . . . .	19
1.3 Scope and Limitations . . . . .	20
1.4 Plan of Development . . . . .	21
<b>2 Literature Review</b>	<b>23</b>
2.1 Introduction . . . . .	23
2.2 Human Motion and Gait . . . . .	24
2.3 Computer Vision . . . . .	24
2.3.1 Computer Vision in Robotics . . . . .	25
2.3.2 New Perspectives from Animal Borne Cameras . . . . .	26

2.3.3	Human Motion Analysis Using Computer Vision . . . . .	26
2.4	Inertial Measurement Units and Sensors . . . . .	27
2.4.1	Global Position System . . . . .	28
2.4.2	Inertial Measurement Units in Robotics . . . . .	28
2.4.3	Human Motion Analysis Using Sensors . . . . .	29
2.5	Mathematical Modelling . . . . .	29
2.5.1	Mathematical Models of the Gait . . . . .	29
2.5.2	Linear Kinematics . . . . .	30
2.5.3	Rotational Matrices . . . . .	31
2.5.4	Kalman Filter and Extended Kalman Filter . . . . .	32
2.6	Natural Solutions for Robotic Shortcomings . . . . .	32
2.7	Conclusion . . . . .	33
<b>3</b>	<b>Methodology</b>	<b>35</b>
3.1	System Design . . . . .	35
3.2	Modelling the Lower Limbs . . . . .	37
3.3	Experimental Details . . . . .	39
3.4	Limitations . . . . .	39
<b>4</b>	<b>Designing the Data Capture System</b>	<b>40</b>
4.1	GoPro Hero Session Camera . . . . .	40
4.2	Camera Mount Design . . . . .	42
4.3	Sony Xperia Z3 Compact Smartphone . . . . .	44
4.3.1	AndroSensor . . . . .	45
4.3.2	SensorLab . . . . .	45

4.3.3	Custom Software . . . . .	46
4.3.4	Final Selection . . . . .	47
4.4	Smartphone Mount Design . . . . .	48
4.5	Critical Point Markers . . . . .	48
<b>5</b>	<b>Processing the Captured Data</b>	<b>50</b>
5.1	Processing the IMU Data . . . . .	50
5.1.1	Obtaining IMU Data . . . . .	51
5.1.2	Synchronizing IMU Data . . . . .	51
5.1.3	Preprocessing IMU Data . . . . .	52
5.1.4	Exporting IMU Data . . . . .	53
5.2	Processing the Video Data . . . . .	54
5.2.1	Obtaining Video Data . . . . .	54
5.2.2	Synchronizing Video Sources . . . . .	55
5.2.3	Cutting Critical Video Data . . . . .	55
5.2.4	Undistorting the Video Data . . . . .	56
5.2.5	Tracking and Exporting Marker Positions . . . . .	57
<b>6</b>	<b>Data Fusion and State Estimation</b>	<b>63</b>
6.1	State Estimation . . . . .	64
6.2	Prediction Equations . . . . .	66
6.3	Measurement Equations - Body . . . . .	67
6.4	Measurement Equations - Lower Limbs . . . . .	67
6.4.1	Euler Matrices . . . . .	67
6.4.2	Forward Kinematics . . . . .	68

6.5	Camera Matrix . . . . .	69
6.6	Q Matrix, R Matrix and Initialization . . . . .	69
6.6.1	Defining the Q Matrix . . . . .	70
6.6.2	Defining the R Matrix . . . . .	70
6.6.3	Choosing Initial States . . . . .	70
6.6.4	Initializing the Covariance Matrix P . . . . .	70
6.7	MATLAB Implementation . . . . .	71
<b>7</b>	<b>Results and Discussion</b>	<b>72</b>
7.1	Results . . . . .	72
7.1.1	System Design Results . . . . .	72
7.1.2	Image Processing Results . . . . .	73
7.1.3	Captured and Processed Data . . . . .	74
7.1.4	EFK Results . . . . .	77
7.2	Discussion . . . . .	78
7.2.1	System Design . . . . .	78
7.2.2	Image Processing . . . . .	78
7.2.3	Captured Data . . . . .	78
7.2.4	System Modelling . . . . .	79
7.2.5	Extended Kalman Filter . . . . .	79
<b>8</b>	<b>Conclusions and Future Work</b>	<b>81</b>
8.1	Conclusion . . . . .	81
8.2	Future work . . . . .	82
	Appendix A: Calibration . . . . .	89

Appendix B: Results Graphs . . . . .	90
Appendix C: CD Guide . . . . .	104
Appendix D: Ethics and ELO Documents . . . . .	105

# List of Figures

1.1 Left: Vicon motion capture system tracking the human gait, from [3], Right: The Xsens MVN motion capture suit, from [5] . . . . .	18
1.2 A cheetah wearing the wearable motion capture system designed by Stocks [7]	19
1.3 Flow chart outlining the report structure . . . . .	22
2.1 Pictures of modern bipedal robots Atlas (left) and Cassie (right) from [18] and [19] respectively . . . . .	24
2.2 Insight into object classification by Tesla, image from [25] . . . . .	25
2.3 Dollphins wearing dorsal mounted cameras, image from [28] . . . . .	26
2.4 Dollphins wearing dorsal mounted cameras, image from [5] . . . . .	27
2.5 Basic kinematic model to demonstrate . . . . .	31
2.6 Different bipedal and quadroped robots created by Boston Dynamics, image from [43] . . . . .	33
3.1 Diagram showing the progression and dependence of the major stages of this project . . . . .	35
3.2 Original wearable motion capture system designed by Stocks, image from [7]	36
3.3 Rigid beam model used to model the lower limbs of a biped . . . . .	38
4.1 3D CAD rendering of the GoPro Hero Session action camera from [50] .	40
4.2 Initial camera enclosure designed by the mechatronics research lab . . . .	42
4.3 Final dual camera enclosure designed by the author . . . . .	42

4.4	Action Mounts chest mount showing the front mounting plate . . . . .	43
4.5	Stabilizer mounted to the rear camera . . . . .	43
4.6	Stabilizer mounted to the rear camera . . . . .	44
4.7	AndroSensor mobile application running on the Sony Xperia Z3 Compact	45
4.8	SensorLab mobile application running on the Sony Xperia Z3 Compact .	46
4.9	Screenshots of the custom software solution . . . . .	47
4.10	Smartphone mounted rigidly to the chest piece of the harness . . . . .	48
4.11	Subject wearing Green and Pink Markers to identify critical points . . .	48
5.1	Diagram showing the progression and dependence of the major stages of sensor data processing in this project . . . . .	50
5.2	Diagram showing the progression and dependence of the major stages of video processing in the project . . . . .	54
5.3	GoPro camera remote control . . . . .	55
5.4	Figure showing the user interface of SVP video editing software . . . . .	55
5.5	Images demonstrating the effects of lens distortion. . . . .	56
5.6	4 Frames from the different cameras combined . . . . .	57
5.7	4 Frames from the different cameras combined . . . . .	58
5.8	Using the dltdv toolbox in MATLAB to export marker coordinates . . .	60
5.9	Figure demonstrating the body frame of reference . . . . .	61
5.10	Figure demonstrating the frame of reference of the smartphone . . . . .	61
5.11	Figure demonstrating the camera frame of reference, image adapted from [12]	62
6.1	Figure showing the interplay of various elements of the KF, adapted from [56]	65
7.1	subject wearing the motion capture system . . . . .	72
7.2	iamge processing things . . . . .	73

7.3	Pre filter accelerometer data . . . . .	74
7.4	Pre filter gyroscope data . . . . .	75
7.5	Pre filter GPS and barometer data . . . . .	76
7.6	Bar graph showing the availability of various measurements as a percentage	77
8.1	IMU, Barometer, and GPS availability . . . . .	91
8.2	Front Left Camera availability . . . . .	92
8.3	Front Right Camera availability . . . . .	93
8.4	Back Left Camera availability . . . . .	94
8.5	Back Right Camera availability . . . . .	95
8.6	Front Left Camera Scatter Plot . . . . .	96
8.7	Front Right Camera Scatter Plot . . . . .	96
8.8	Back Left Camera Scatter Plot . . . . .	97
8.9	Back Right Camera Scatter Plot . . . . .	97
8.10	EKF Configuration 1 - Actual State Values . . . . .	98
8.11	EKF Configuration 1 - Estimated State Values . . . . .	99
8.12	EKF Configuration 2 - Actual State Values . . . . .	100
8.13	EKF Configuration 2 - Estimated State Values . . . . .	101
8.14	EKF Configuration 3 - Actual State Values . . . . .	102
8.15	EKF Configuration 3 - Estimated State Values . . . . .	103
8.16	Ethics clearance . . . . .	105
8.17	ELO clearance . . . . .	106

# List of Tables

3.1	Known design elements of the project . . . . .	36
4.1	Possible frame rate and resolution combinations on the GPHS camera . .	41
4.2	This table shows the different headings of the output AndroSensor file . .	47
5.1	Table showing the different units of the output AndroSensor file . . . .	51
5.2	Table showing the relation between points and markers . . . . .	60
6.1	Table showing the different states of the model to be determined by the EKF . . . . .	64

# Glossary

Abbreviations listed here are used throughout the document.

- DOF - Degrees Of Freedom
- GPHS - GoPro Hero Session
- MEMS - Micro Electromechanical System
- CSV - Comma Separated Values
- 3D - Three Dimensional
- IMU - Inertial Measurement Unit
- KF - Kalman Filter
- EKF - Extended Kalman Filter
- CAD - Computer Aided Design
- SVP - Sony Vegas Pro
- CD - Compact Disc
- MP4 - MPEG-4 Part 14 Media Container
- AVI - Audio Video Interleaved Media Container
- RGB - Red Green Blue

# Symbols

Mathematical Symbols listed here are used throughout the document.

- $\phi$  - Roll
- $\theta$  - Pitch
- $\psi$  - Yaw
- $x$  - positional coordinate of the x-axis in the inertial frame
- $y$  - positional coordinate of the y-axis in the inertial frame
- $z$  - positional coordinate of the z-axis in the inertial frame
- $X$  - State Vector of the System
- $Q$  - Process Noise Matrix
- $R$  - Measurement Noise Matrix
- $F$  - State Transition Matrix
- $H$  - Observation Matrix
- $P$  - Covariance Matrix
- $K$  - Kalman Gain

# Chapter 1

## Introduction

### 1.1 Background to the Study

Human motion capture systems are often very costly and confine the capture area to a certain confined space. These limitations prevent us from understanding bipedal motion in complex environments, knowledge that proves to be critical in the development of functional humanoid robotics. Examples of such systems can be seen in [1] where 8 cameras and stereo vision was used to recreate a 3D model of a walking person and in [2] where 10 Vicon cameras and a full body marked suit was used for the same objective.

A popular approach taken to overcome the spatial limitations inherent in camera based systems is the use of body mounted sensor networks. A commercial sensor suit developed by XSens and demonstrated in [3] can accurately recreate a 3D model of human motion using inertial sensors and data fusion. Similar work completed by Seel et al. [4] further proves the accuracy of such systems. Although the mentioned systems overcome the stated limitations; they do so at increased complexity and cost. They often lack modularity and their proprietary software is difficult to adapt.



Figure 1.1: Left: Vicon motion capture system tracking the human gait, from [3], Right: The Xsens MVN motion capture suit, from [5]

Recent work [6] completed by the Mechatronics Lab at the University of Cape Town showed data capture with both subject-borne cameras and sensors can be used to better understand unconstrained movement in a natural environment. This was based on research completed by Stocks [7] at the same laboratory. The presented work showed the successful kinematic modelling of a cheetah (*Acinonyx jubatus*) tail whilst running freely. The importance of understanding motion in the natural world is outlined in [8] and is the cornerstone of biomimicry as defined by [9].

## 1.2 Objectives of the Study

Depth imagery in the field of human motion capture has been extensively reviewed in [10], where the lack of data from complex movements in different environments is listed as a challenge. This reaffirms the difficulty stated in the previous section. Solely relying on motion sensors to understand the gait has been reviewed by [11]. Although this approach was found to be accurate for external environments it has limitation with respect to cost and sensor disturbance. From these reviews it is clear that a middle ground must exist that can combine the strengths of both the approaches to provide a holistic solution.

This research project aims to show that subject-borne sensors, primarily a combination of cameras and IMUs, can provide researchers in the field of health sciences, biomechanics and biomimicry with extensive datasets to better understand and model the bipedal motion of humans. It builds on the foundational work presented by Stocks and Patel [7] and envisions to implement a similar system to track data points on the lower limbs of a runner. The original prototype system is shown below, mounted to the back of a cheetah.

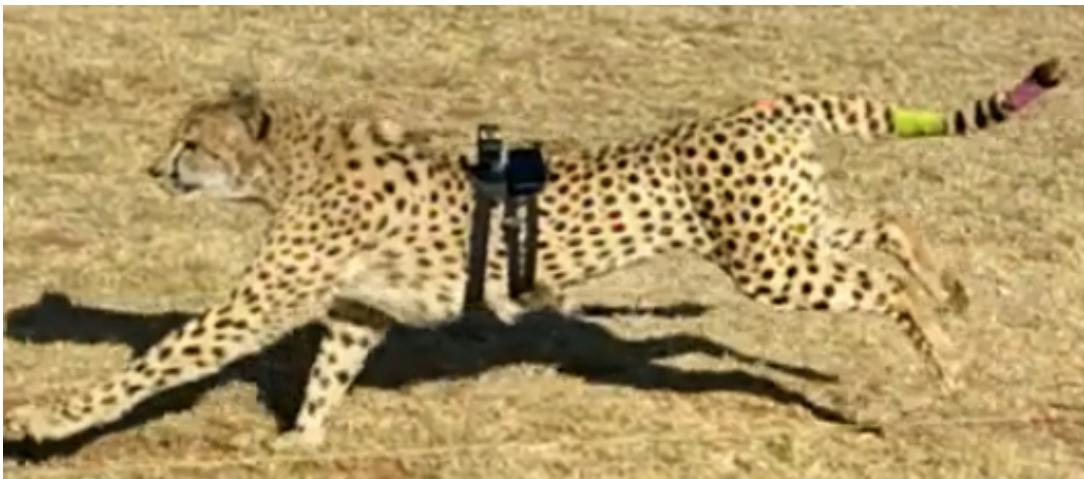


Figure 1.2: A cheetah wearing the wearable motion capture system designed by Stocks [7]

This research project firstly aims to design such a data capture system using available hardware solutions and basic 3D printable components. By using such elements the system gains modularity whilst the overall cost and development time is reduced. This rapid prototyping philosophy was chosen such that the project could be completed within a 4 month time frame.

While developing and constructing the above mentioned system a basic model of the human lower limbs will be created. The model will need well defined mathematical definitions such that kinematic principles can be applied to quantify critical gait elements. Important simplifications and shortcomings of the model need to be explored and methods to minimize their effects created.

The design and software implementation of a system to extract, process and fuse the various data sources needs to be developed. This implementation will include, but is not limited to, an Extended Kalman filter and an image processing algorithm to identify critical points in the video data.

Finally this thesis will pay close attention to similarities in understanding the human gait using technologies used to control the dynamics of bipedal humanoid robots. Since these robots attempt to imitate human motion many of they key principles in the field of robotics relates directly to to the bio-mechanics of humans.

### 1.3 Scope and Limitations

The scope of this research is to model and estimate the human lower limbs during a flat ground steady state run. The motion will be estimated by determining the different joint angles as well as the motion of the runner w.r.t. the inertial frame. Joint angles are a popular method of quantization and the availability of rich datasets allow comparative analysis on the final findings of this work. This is the first logical step in the iterative design process to eventually understand movement in complex environments using wearable motion capture systems.

The availability of equipment also influenced the quality and accuracy of the system. Due to financial constraints the system was designed to use available hardware and minimize additional expenses. Although the the system designed cannot be classified as low cost, when compared to other methods of motion capture it becomes financially attractive. The components selected are also interchangeable and need not exactly match the specifications presented herein.

It should be noted that the research presented herein does not seek to push the boundaries of modern sensor technology, nor does it wish to re-imagine understood and accepted models of natural phenomena. Instead, a methodology is proposed that brings together elements from exciting disciplines of research such that richer datasets can be generated and studied. This research therefore serves as a proof of concept for a novel wearable motion capture technology.

There is another distinction to make with regard to scope of the project; that is the distinction between kinematics and dynamics. This project is aimed at understanding, modelling and estimating the kinematics of lower limbs, that is to say the movement and motion of the lower limbs but not the forces and torques causing them. These forces are important elements of motion, but require some adaptation to the proposed methodology to fully understand.

Furthermore this thesis will not compare different running styles or comment on their energy efficiency. Interpreting and rehabilitating the human gait is best left to medical professionals. This system may be applicable to work done in such professions, but the task of developing this system is one of engineering.

Finally a large portion of this project relies on software written for MathWorks' MATLAB [12]. This software is single purpose and serves only this thesis. The software itself is not meant to be modular or generalized, yet can serve as a guideline for research using a similar methodology. The software can be found on the accompanying disc. Some software snippets of critical importance has been added to this thesis to highlight the important aspects of implementing the various mathematical constructs. The various Dassault Systmes SOLIDWORKS [13] models are also present on the same disc. These models are also specific to this thesis and can only provide insight for adaptations.

## 1.4 Plan of Development

The following chapter contains an extensive literature review where various methods of modelling and verifying the human gait has been discussed. There are also sections dedicated to subject borne data capture, computer vision, inertial measurement units (motion sensors), humanoid robotics and mathematical modelling.

This is followed by a chapter titled methodology that presents the planning and ideation of the thesis. It serves as a link between the theoretical work presented in the literature review and the engineering approach and application detailed in the chapters that follow it. It lays out a plan and shows how engineering specifications were generated

from a generally defined problem.

The final three chapters that make up the body of this report are titled "Designing the Data Capture System", "Processing the Captured Data" and "Data Fusion and State Estimation" in order of appearance. True to their title they present the processes followed to complete the major milestones of the project.

In closing a chapter is dedicated to presenting and discussing the results obtained, followed by the final chapter that draws conclusions from the presented work and makes recommendations on future work. The following flow diagram summarizes the progression of this report.

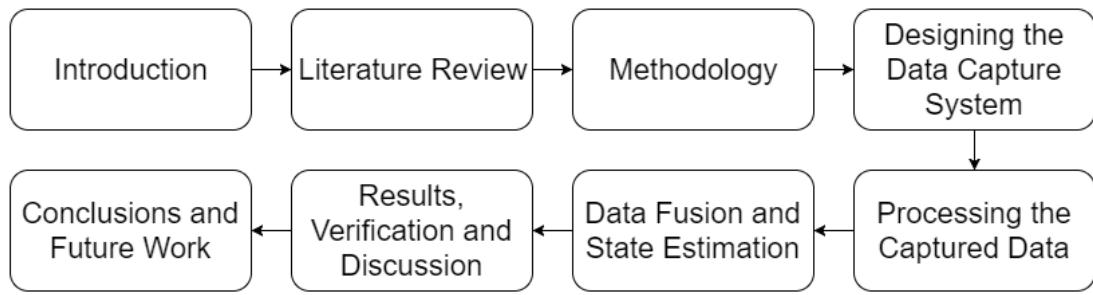


Figure 1.3: Flow chart outlining the report structure

The end matter of this report consists of 4 different appendices. Appendix A discusses the calibration of the hardware and Appendix B contains graphs relating to the results section of this report. This is followed by Appendix C that details the accompanying CD and Appendix D that contains the Ethics and ELO clearance of this project.

# Chapter 2

## Literature Review

This section reviews various academic sources related to the methodology proposed. It will look at the various fields of engineering as well as bio-mechanics, biomimicry and applied mathematics to form a holistic understanding of the design space. Importantly the technologies discussed will also be related to the field of robotics as many concepts transfer easily from bipedal humans to bipedal robots.

### 2.1 Introduction

This research project brings together various disciplines of research. By combining techniques from computer vision, sensors and data fusion we can design and develop new way of capturing human gait data. Whilst the fields of biomimicry and bio-inspired robotics are relatively new, recent advances in related fields such as artificial intelligence and robotics have invigorated the pursuit of functional humanoid robotics.

Kaneko et al. described various components of humanoid robotics in [14]. Herein a fundamental element of dynamics is discussed and improvements to the robots mobility outlined as the first step in the iterative design process. The same author published work [15] relating to a functional leg module to be used for such robotic projects. These works shows that engineers have been trying to replicate the bipedal motion of humans for some time wit relatively limited success.

If we observe some of the worlds foremost attempts at bipedal robotics such as Boston Dynamics' Atlas [16] and Agility Robotics' Cassie [17] we can see that recent attempts are improving rapidly. This thesis believes that with bigger datasets of human motion in complex environments we can better design and control robotic lower limbs.

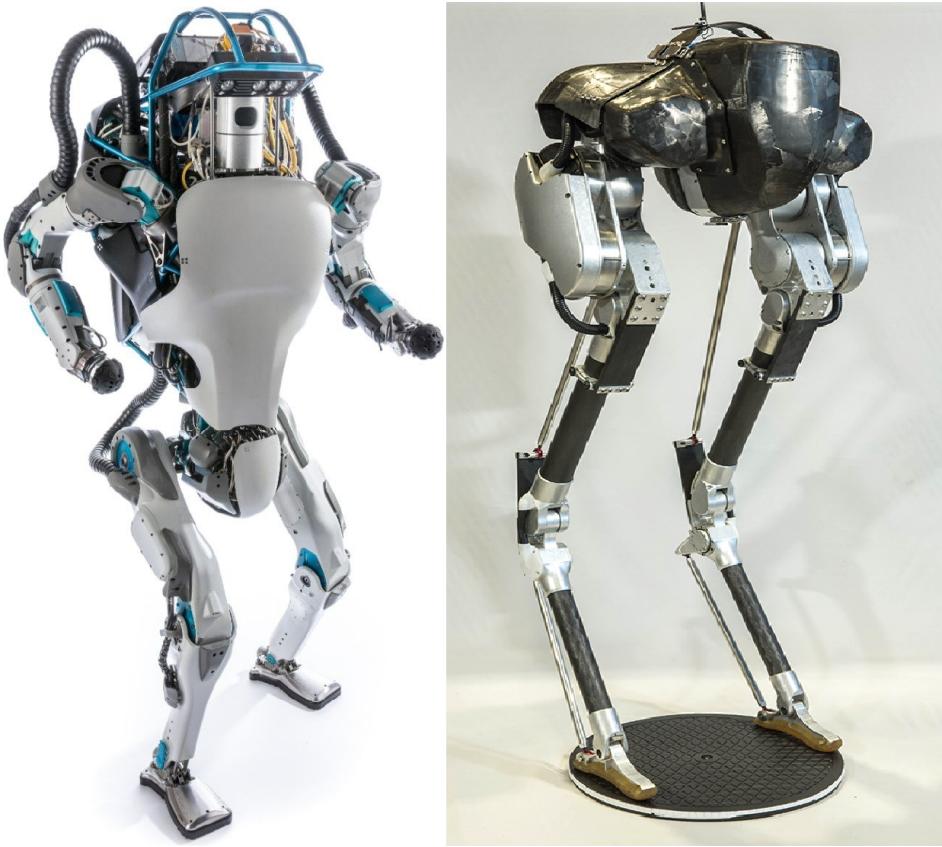


Figure 2.1: Pictures of modern bipedal robots Atlas (left) and Cassie (right) from [18] and [19] respectively

## 2.2 Human Motion and Gait

The human gait is well understood and has been studied in detail as it is a fundamental part of human mobility. It is one of the first skills developed in infancy and its importance for healthy development, as outlined by Adolph et al. [20], cannot be understated. Walking and running are also critical factors in transportation and geographical movement of people and goods in developing countries where public transport is underdeveloped and private transport not within the means of the populous. Finally walking and running as exercise has proven benefits as shown in [21] (general health) and [22] (mental health). There is thus clear evidence that the human gait has earned its right as a field of study in academia.

## 2.3 Computer Vision

While the previous section answers "why" understanding the human gait is important, the following sections will explain fields that contribute to the question of "how" the

gait is studied. The technologies and methods used to quantify it. This section titled computer vision should be interpreted within the context of this document. It will be used interchangeable with image processing as the underlying philosophies of both methodologies are algorithmic interpretation of images.

Image processing as a field was born from digital signal processing as it relates to the extraction of critical data from noisy data streams. Computer vision is the use of computational methods to achieve the same end goal; ultimately trying to emulate human vision. Image Processing has many different approaches and methodologies. These include but are not limited to feature detection, pattern recognition and classification. Modern works such as [23] and [24] have shown how neural networks perform previously mentioned tasks with unprecedented accuracy.

### 2.3.1 Computer Vision in Robotics

Recent improvements to real time image processing has allowed amazing technological breakthroughs in fields closely related to robotics. One such breakthrough is the rapid improvement of self-driving cars developed by Tesla. These vehicles use vision based technologies and real time image processing to navigate complex and changing road networks. The figure below shows how a Tesla identifies different roadside artefacts.



Figure 2.2: Insight into object classification by Tesla, image from [25]

Another interesting use for computer vision in robotics is the ability to identify and classify real world objects. This allows for robotics to perform menial household tasks. Finally work completed by Taylor et al. [26] shows the importance in computer vision in robotics assisted surgery. Computer vision is an important step in automatizing robotics and due to the rapid progression in artificial intelligence, it is a field with a large potential

for growth.

### 2.3.2 New Perspectives from Animal Borne Cameras

Patel et al. [6] showed that using animal borne cameras and motion sensors, the tail kinematics of the cheetah (*Acinonyx Jubatus*) could be tracked. Patel's work was partly inspired by Kane et al. [27] where falcon (*Falco Peregrinus*) borne cameras were used to better understand airborne pursuit of prey. Giving researchers a new perspective on the behaviour of animals in the natural world.

Further work completed by Pearson et al. [28] showed that cameras mounted to dolphins (*Lagenorhynchus Obscurus*) could provide insight into their movement, social and foraging strategies. Using cameras to study ocean-life has become a popular methodology in recent time due to difficulties imposed by their environment. In essence we struggle to understand flying and swimming animals due to their complex environments. The following image shows how these devices are carried by various dolphins.



Figure 2.3: Dolphins wearing dorsal mounted cameras, image from [28]

The above research has shown the unique benefits of having cameras and sensors mounted to the subject in question.

### 2.3.3 Human Motion Analysis Using Computer Vision

From Chen et al. [10] using depth imagery to understand human motion we can see that this is a popular technique. Because imaging is a popular method in medical research imaging various human movements has a large set of well established methodologies. Naturally this has formed a foundation of using cameras to capture human movement. Companies like Vicon, Optitrack and Motion Analysis have created multiple consumer products for quantifying motion using cameras. The following image shows a typical motion capture setup using multiple cameras.

One system often used for motion capture is the Microsoft developed Kinect. [29] [30] [31] have shown positive results in modelling and quantifying the human gait using this technology. Open source software like OpenKinect allows for easy implementation and



Figure 2.4: Dollphins wearing dorsal mounted cameras, image from [5]

configuration. A drawback to this methodology is that all of these works require controlled environments due to the nature of the technology.

A thorough search for wearable system using cameras to identify critical points on the lower limbs was done, but no pre-existing research was found. It was concluded that this thesis is novel and exploring a new approach to understanding lower limb kinematics.

## 2.4 Inertial Measurement Units and Sensors

IMU's are a staple of electrical engineering as applied to dynamic systems. These sensors give us insight as to how an object is moving in space by providing data relating to orientation and acceleration of said system. These data points are created by electronically interpreting signals generated by micro-electromechanical system (MEMS). Modern smartphones have built in IMU's that are not only accurate [32], but also easy to interface with due to the open source nature of the Android operating system [33].

Generally Smartphones contain the following sensors:

- Accelerometer
- Gyroscope

- Magnetometer
- Barometer
- Temperature

**Accelerometers** provide linear acceleration data; these accelerations may be constant (eg. gravity) or changing (eg. relative motion). In smartphones they are usually based on MEMS that use various mechanical phenomena to determine motion.

**Gyroscopes** provide rotational data of the sensor relative to the inertial frame. These sensors generate angular velocity data by using .

**Magnetometers** provide information relating to the macroscopic magnetic fields in a certain area. These sensors can measure the direction, strength, or relative change of fields in three different dimensions relative to the smartphone.

**Barometers** are finely tuned atmospheric pressure sensors that can determine pressure an object is experiencing. By combining this pressure data with a well defined map of different pressure the the relative height with respect to sea-level can be calculated.

**Temperature** sensors generate local temperature data of the surrounding environment. They are important in smartphones that use lithium ion or lithium polymer batteries that can explode at high temperatures.

These sensors can be used together to better model the position, velocity and acceleration of a modern smartphone. This is easily seen when a smartphone rotates the display when held in landscape.

### 2.4.1 Global Position System

GPS (Global Position System) is a space based navigational system that uses satellites to determine a receivers absolute position on earth. This system was developed by the United States Air Force ion 1973 and made available for public use in the 1980s. It has since been improved by the addition of satellites.

### 2.4.2 Inertial Measurement Units in Robotics

IMUs are integral in the functioning of robotics. Up until very recently intelligent robotic systems had no sense of vision to provide feedback for their internal control systems.

Instead this feedback was generated by various sensors providing information about the dynamics of these systems. [15] showed the importance of feedback to control bipedal robotic lower limbs. This feedback is achieved with different electronics components including gyroscopes and accelerometers; they are preferred above potentiometers since they do not mechanically intrude on the system.

Another common application for sensors in robots is for the use of navigation. By using accurate IMU's mounted to the body of drones etc pose estimation can be used to control movement.

### 2.4.3 Human Motion Analysis Using Sensors

Picerno completed an extensive review of motion sensor based data capture for human motion in [11].

Some new methods using interesting sensors have been developed to log human motion. [34] showed that by using highly sensitive strain sensors positioned on various joints the movement of such points of interest could be quantified. Another exotic method is the use of soft carbon nanotube capacitive sensors as in [35]. These sensors are flexible and non intrusive allowing comfortable data capture.

A low cost approach in the form of a smartphone and wrist mounted sensors was used by [36] to show alternative methods to interpret human arms movements. Finally software developments by [37] has allowed for more accurate simulations to be produced using inertial sensors. These papers are recent and shows that modern technologies and approaches to capturing motion data are being developed.

## 2.5 Mathematical Modelling

The binding element presented in this work is the underlying mathematics. Using various mathematical tools and methods known to robotics and bio-mechanics it is possible to transform various data types in various frames of reference to a singular model.

### 2.5.1 Mathematical Models of the Gait

Before exploring complex methods and tools used to analyze the human gait, it is important to select and understand the model they are derived from. Due to the large

amount of existing research related to the human gait some models have been well established. These models are capable of quantifying important elements of the human gait such as gait period, dynamic joint forces and neuromuscular control.

Some fundamental work complete by Zajac, Neptune and Winters will be discussed to better understand existing models.

In work completed by Zajac et al. [38] it is interesting to note the how modelling difficulties are compared to that seen in bipedal robotics. In this work he also places some important bounds on human joints. He argues that the maximum DOF (Degrees of Freedom) that any single joint can have is 6; 3 for translation and 3 for rotation. He also constrains body segments as rigid and that the internal happenings of a body segment is insignificant.

In further papers published by these authors [39] and [40] various dynamic simulations are tested against proposed methods and subject studies. These papers confirm the multi rigid segment model for studying human dynamics. Since this study is only concerned with the kinetics the assumption can be made that the model is adequate in kinematic analysis.

### 2.5.2 Linear Kinematics

By using kinematics we can quantify and understand the movement of the lower limbs. Kinematics is a branch of mechanics that fully defines the motion of a point with respect to position, velocity and acceleration (be it linear, rotational or a combination). Kinematics does not however describe the forces, torques or other variables that may affect that point. This is due to a fundamental assumption in kinematics that the point is massless.

Kinematics can be broken up into 2 main branches: *forward* and *inverse*. To illustrate the matter the following diagram is that of a basic kinematic model.

In this figure the position of point P is defined by 2 lengths, L1 and L2 with different lengths and angles from a set of shared axis. In forward kinematics we can find P if we know the angles and lengths of the different links in the system. The motion of point P can then be described by looking at how the angles and lengths of the links in the system change over time.

Inverse kinematics uses knowledge of different points in the system, such as the origin and P and the lengths of the links in the system to determine the angular offsets of each length. Since these produce a set of linear equations, the more unknowns we are faced with the more possible solutions we can generate.

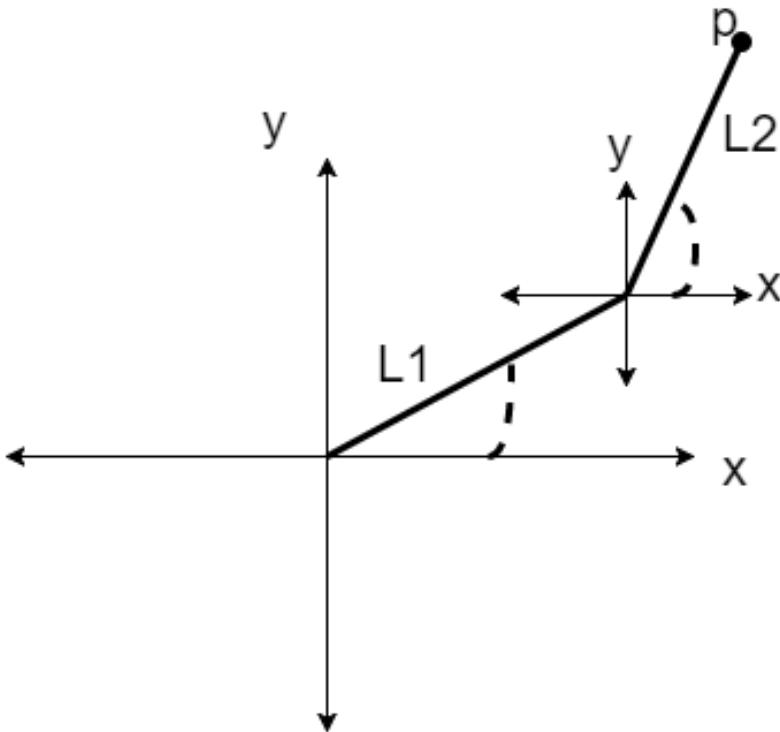


Figure 2.5: Basic kinematic model to demonstrate

As discussed in the previous section a common method of modelling the human lower limbs is to use a collection of rigid beams.

### 2.5.3 Rotational Matrices

There is an underlying difficulty in mathematically fusing various data sources and models; that of finding a common frame of reference. With the intent of using Lagrangian mechanics good definitions for the different frames are critical. This thesis will primarily use 2 different frames of reference. The inertial frame and the body frame.

The inertial frame (or world frame) can be defined in different ways as seen in [41], for the purpose of this study the NED (North East Down) definition is used. This configuration is also known as the local tangent plain and is often used in aviation.

Rotational matrices are mathematical objects that rotate vectors in three dimensional space. Since most engineering is constrained to the physical three dimensional world these matrices commonly rotate 3 dimensional vectors with a 3x3 sized matrix.

### 2.5.4 Kalman Filter and Extended Kalman Filter

The Kalman filter is a mathematical tool used to estimate the states of a system. All measurements contain some unwanted elements of noise that produce uncertainty. Another source of uncertainty is the imprecision in the model. Simplifying assumptions disregard the minute details that when summed can have an effect on the interpretation of the data. To minimize these uncertainties it is important to filter the datasets correctly. Fortunately, estimation can be used as a form of filtering to reduce the impact of these uncertainties.

Another powerful element of the Kalman filter is its ability to fuse data from different sources to compute a more holistic picture of the underlying system. Fusion allows us to interpret sensor data within constraints of other sensors, creating a more accurate dataset. For example we can negate the drift of an accelerometer if we have absolute positional data provided by a GPS.

There is also an important distinction to be made between the KF and the EKF. To briefly explain this it should be understood that the Kalman filter was the original concept as developed by Rudolf E. Klmn and he EKF the extension of said work. The KF has an inherent limitation that it can only be applied to linear systems, whereas the EKF can be applied to non-linear system operating within a certain defined range.

The KF itself can be broken down into 2 fundamental stages of operation; a prediction stage and update stage. The prediction stage takes the known current states of the system and estimates what the measurements should be for the next time interval. The measurement stage takes in current measurements and mathematically determines the states. The states of the system are user defined parameters that can often not be directly measured.

## 2.6 Natural Solutions for Robotic Shortcomings

Naturally the question arises: why would we want to better understand the dynamics of animals? A persistent problem in the field of modern robotics is that of mobility; robots struggle to navigate real world surfaces and obstacles. Work by Patel et al. [42] shows how we can look towards nature for inspiration to solve this mobility problem.

As demonstrated by various prototype robots built by Boston Dynamics bipedal robots are severely limited in manoeuvrability when compared to animals. This is due to the

longest iterative design process known to man, evolution. Pictured below is a collection of bio-inspired robots build by Boston Dynamics.



Figure 2.6: Different bipedal and quadroped robots created by Boston Dynamics, image from [43]

## 2.7 Conclusion

This chapter has shown the direct parallels of technologies related to gate capture to dynamic robotic systems. With these strong parallels in mind the transferability of these systems from humans to humanoid robots is clear. In the same manner we are able to take a bio-mechanical look at the human body and treat it as a dynamics mechanical system instead of the complex bio-chemical and physiological system it really is.

This technique of abstracting systems to different domains of knowledge allows us to apply engineering methods and design to complex problem spaces. As mechanical engineers have used resistive networks to understand thermodynamics [44] and control engineers have used mechanical models and electrical models interchangeably to apply control principles [45], there is power in this methodology. Fusing different methods from different fields has proven its usefulness and this work will use this approach of horizontal thinking to create its own unique methodology.

As discussed in this chapter the importance of understanding the human gait cannot be understated. The recent breakthroughs in computer vision and neural networks has reignited a field that has potential to truly change our day to day life. The ever increasing ability of sensors technology and data capture systems allow us to quantify what we have never been able to and the underlying fundamental mathematical methods never seem to fall short.

The future of humanoid robotics sits at the overlap of computer vision, IMUs and biomimicry; add to this some form of general intelligence and the world reaches a stage of automation and transformation only imagined by authors such as Asimov and Wiener. Perhaps this work could contribute to that future.

# Chapter 3

## Methodology

To ensure the success of this project a basic plan of action was created. The following diagram shows the critical phases of the project, as interpreted from the terms of reference.

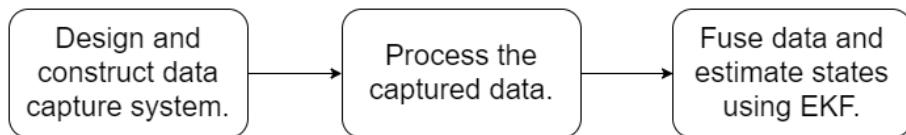


Figure 3.1: Diagram showing the progression and dependence of the major stages of this project

Due to the availability of equipment, financial limitations and time constraints various design parameters were predetermined. These known elements were central to the design process.

### 3.1 System Design

This section is dedicated to defining and understanding the specifications of the data capture system. Due to the lack of existing literature on wearable motion capture systems a starting point had to be determined. This point would serve as the first iteration of an iterative design process to fulfill all defined requirements.

Work completed by Stocks [7] was used as this starting point. The original system comprised of 2 GoPro Hero Session cameras rigidly connected to a Sony smartphone acting as the main sensor. This was all mounted to a GoPro Fetch harness [46] such that it could be carried by various quadrupeds. The original system is pictured below.



Figure 3.2: Original wearable motion capture system designed by Stocks, image from [7]

The system was adapted to use 4 GoPro Hero Session cameras and an IMU mounted to the torso of the subject. Two of the four cameras mounted to the chest of the subject and the other two mounted to the middle-back of the subject.

The cameras will record the lower limbs of the subject while the IMU will log inertial data from the body of the subject. By having cameras on both the chest and back of the subject, the front and rear elements of the gait can be captured. The video data from the cameras will provide information about the kinematics of the lower limbs with respect to the cameras while the IMU will provide motion data of the body with respect to the inertial frame.

The following equipment was provided by the Mechatronics Research Lab and was chosen as the main components to use in the system

Item	Selected Equipment	From
Camera	4 GoPro Hero Session Cameras	[47]
IMU	1 Sony Xperia Z3 Compact	[48]
Chest Mount	1 Action Mount Chest Mount	[49]

Table 3.1: Known design elements of the project

The problem space and concept design of the system has been outlined above, but to further the design process designable specifications need to be defined. By looking at the various limitations of existing systems and the available components the following specifications were identified.

- Two stereo housings to hold the cameras
- Housing to hold the smartphone
- Chest mount to hold the cameras and IMU
- Connecting hardware to mount camera housing to the chest harness
- Cameras must be stable during running
- IMU must be rigidly mounted to front cameras
- Cameras must capture full lower limb motion
- Harness must be comfortable during running
- Harness must not impede natural gait of subject
- Harness must be as light as possible
- Harness must fit different size torsos
- Harness must be unisex
- System must be remote controlled as far as possible

These specifications ensure a system that can be used by a large demographic of people. This allows the system to be used in a variety of applications from engineering to medical science. The system can also be used by subjects with prosthesis as it requires only markers to be attached to the lower limbs. This is extremely useful in the rehabilitation of amputee subjects.

## 3.2 Modelling the Lower Limbs

To interpret the data and the underlying mathematics a model of the human torso and lower limbs must be created. This model consists of the lower limbs being represented as rigid links. Each leg is comprised of three different links: a thigh, calf and foot. The joints

connecting the links have limited degrees of freedom to simplify the model. The ankle (serving as the joint between the foot and calf) is assumed to have a single rotational degree of freedom (pitch). The knee (serving as a joint between the calf and the thigh) has also been limited to only have a pitch element. Finally the hip (joining the thigh to the body) has been given 2 rotational degrees of freedom: pitch and yaw. This rigid model can be seen in the following figure.

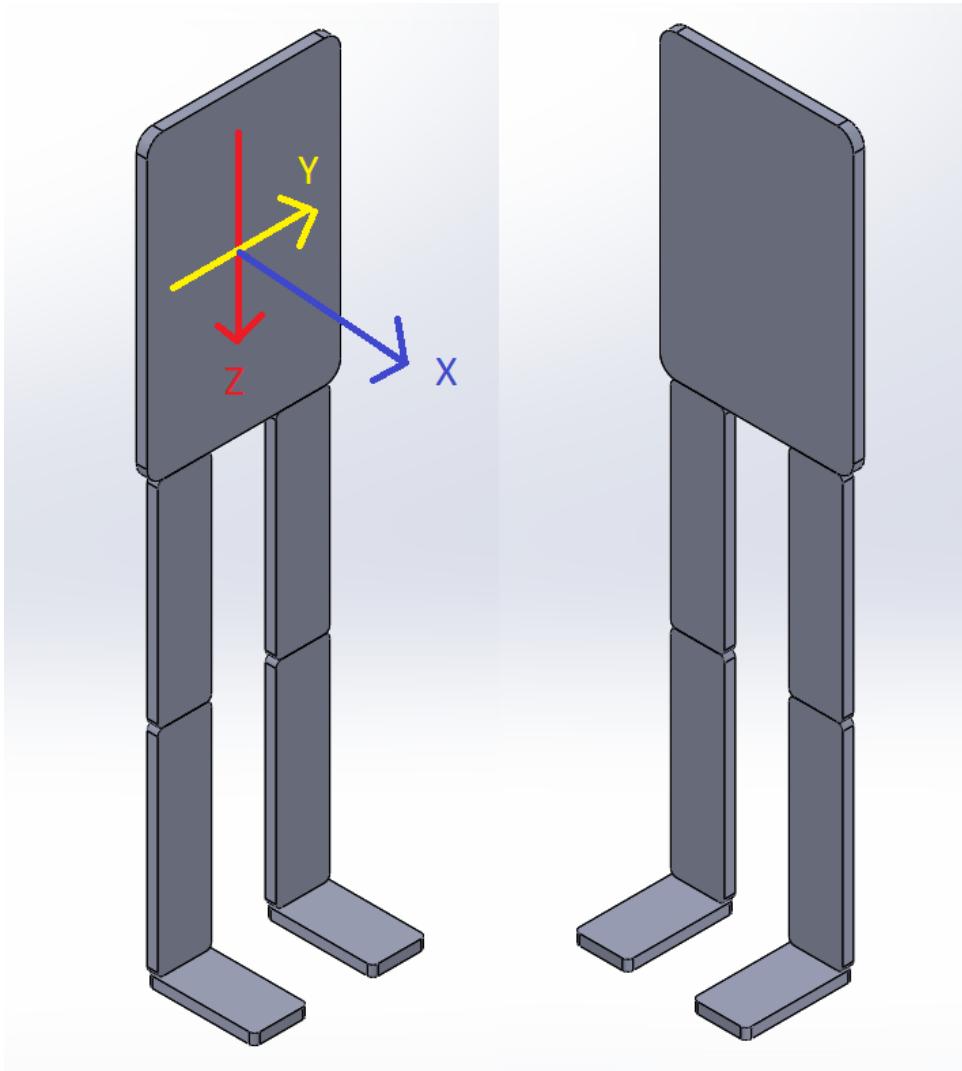


Figure 3.3: Rigid beam model used to model the lower limbs of a biped

A very important benefit to using such a model is the transferability to a robotic implementation of the lower limbs. Often robot joints are implemented using a variety of DC machines or servomotors. By limiting the freedom of various joints we simplify the joints and reduce the amount of servomotors required to mimic the model. This allows for both cheaper and less complex bipedal robotic limbs to be constructed.

### 3.3 Experimental Details

The data was captured during a short straight road run where the runner started from a standing position and accelerated to a steady state running pace. This allowed us to capture some transients (accelerations) from the run that can be used to initialize the proposed EKF. The run lasted about 18 seconds providing us with 1800 datapoints. The test was performed 3 times and a final dataset was chosen after examining the data gathered.

### 3.4 Limitations

The scope of this research does not include runs over rough terrain as this is the logical next step after flat ground steady state running is modelled and estimated correctly. It is assumed that given robust EKF and complete image processing solution the system would be transferable for running on various terrains.

This research will also not analyse the running style, efficiency or balance of the subjects, since this thesis concerned with the engineering involved in creating a novel data capture system.

Another source of limitations is the nature of the rigid beam model. This non elastic model does not take in to account the slight compression and expansion of the limbs during running. The model also omits yaw and roll parameters about the knee and ankle as well as roll parameters about the hip; reducing the overall completeness of the model for the sake of simplicity. Finally the model assumes a rigid stationary chest that introduces some error relating the camera data.

In steady state running the chest will rotate opposite to the hips to balance the angular momentum of the runner. Since the chest swings at a rate proportional to the gait period this could be modelled as a harmonic oscillator. This idea was explored in [6] where the cheetah spine was partly modelled in this fashion with positive results. It would also be possible to interpret the rotational rates of the chest from the IMU data.

Various limitations are inherent in this project due to its novelty. These limitations have been noted such that future work could improve upon certain elements while this foundational work serves as a point of reference.

# Chapter 4

## Designing the Data Capture System

To obtain data for the Extended Kalman Filter, a data-capture system needed to be designed. Since the data sources have been identified as multiple video streams and various sensors from a smartphone these devices must be mounted to a wearable harness. The following specifications are highly modular as any camera source and any smartphone with sufficient sensors and sufficient data capture rate would suffice.

### 4.1 GoPro Hero Session Camera

Due to the availability of GoPro Hero Session cameras the wearable motion capture system was designed with these in mind. These cameras only take up a volume of  $250cm^3$  and have a square housing measuring  $6.3cm$  on all sides. The following figure presents a 3D rendered model of the camera.

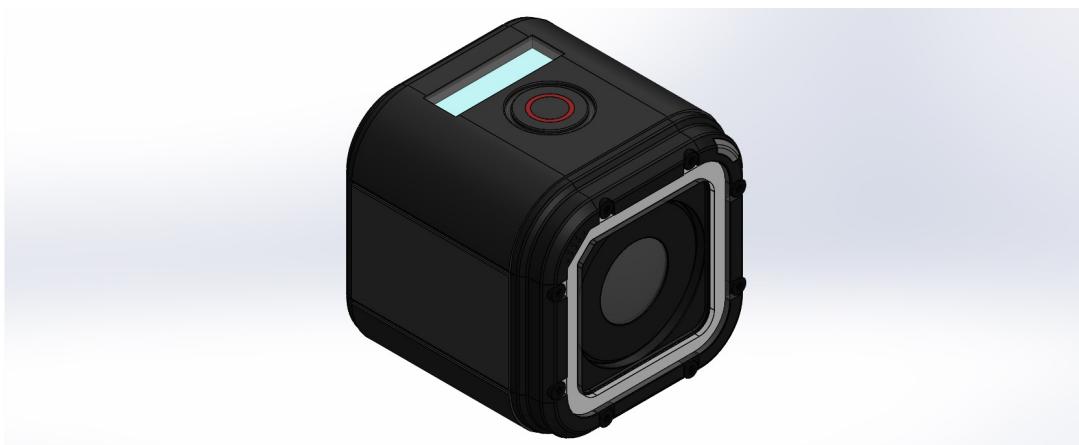


Figure 4.1: 3D CAD rendering of the GoPro Hero Session action camera from [50]

The camera can capture videos at a variety of frame rates and a variety of resolutions. These settings are limited as the software that controls the camera is proprietary. The set frame rates and resolutions are presented in the table below.

Resolution	Frame Rates
1920 x 1440	30 fps, 25 fps
1920 x 1080	60 fps, 50 fps, 48 fps, 30 fps, 25 fps
1280 x 960	60 fps, 50 fps, 30 fps, 25 fps
1280 x 720	100 fps, 60 fps, 50 fps, 30 fps, 25 fps
848 x 480	120 fps, 100 fps

Table 4.1: Possible frame rate and resolution combinations on the GPHS camera

The relative motion of the lower limbs appeared to move rapidly when viewing test footage and therefore the highest possible frame rate with the best resolution was chosen. The camera was configured to record at 100Hz and at a resolution of 1280 x 720 pixels. This was chosen as the quality of the 848 x 480 video was too distorted to identify the marker centres with computer algorithms.

The camera also has the ability to record using a normal lens or a wide angle lens. The field of view of the camera greatly increases with the wide angle lens, but its focal length decreases proportionally. The wide lens also produces more distortion when compared to the normal lens as explained in [51]. Due to the relatively narrow area of capture needed the camera was configured to use the normal lens as it would decrease distortion and increase local length, without compromising the area of interest.

Naturally, working with proprietary hardware presents some difficulties. One of these difficulties is created by the GoPro camera regulating its exposure automatically. In darker environments the GoPro will automatically change to low light mode, causing inconsistent light levels in videos taken in different conditions. This causes uncertainty when trying to use feature detection since the varying light levels change the relative colors of the markers.

Another difficulty is due to the output video files generated by the GoPro. These files have a .MP4 file extension implying that they have already been compressed. This compression causes a loss of precision opposed to the raw video data being recorded. Compression had been implemented to save memory on the GoPro's micro SD memory card. Decompressing this video data will be discussed in the following chapter.

## 4.2 Camera Mount Design

Some initial work on modelling a housing for the camera was completed by the Mechatronics Research Lab. This was a 2 part 3D printable enclosure with no mounting points and limited control access. The enclosure is pictured below.

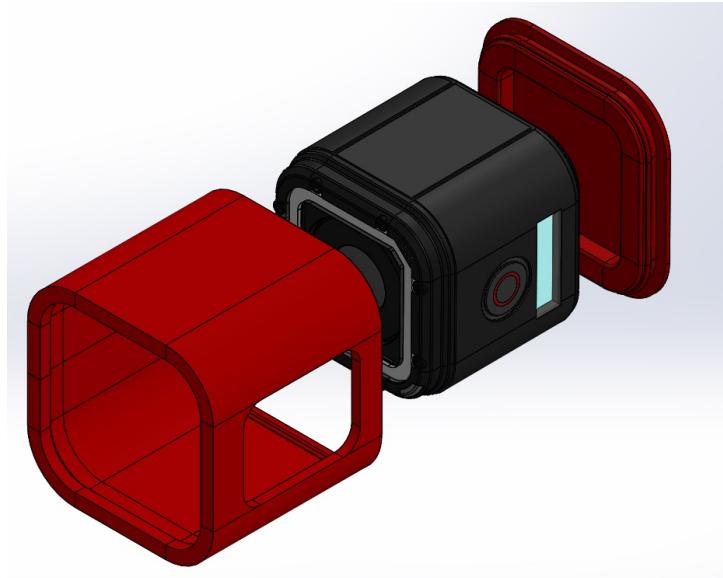


Figure 4.2: Initial camera enclosure designed by the mechatronics research lab

This model was heavily modified using Dassault Systems SOLIDWORKS software to enclose 2 cameras mounted side by side. The bracket also needed a mounting point to join to the chest mount. Finally the bracket needed to be lightweight, provide access to the camera controls and not obscure the built in status screen of the cameras. Various cut-outs were made to the frame sides. The following figure shows the final dual camera bracket that was 3D printed. This bracket can be found, along with other design iterations, on the accompanying CD.

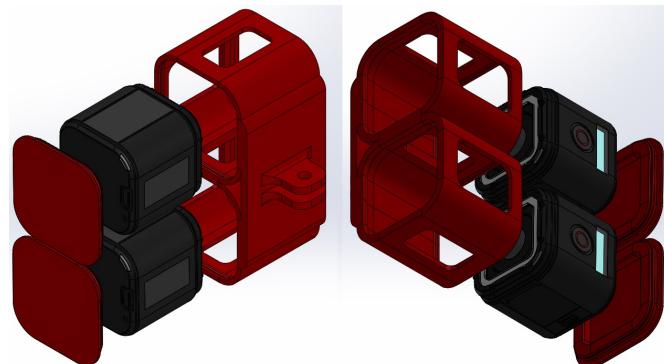


Figure 4.3: Final dual camera enclosure designed by the author

As seen in the figure above the sides of the housing was opened to reduce the overall weight of the mount. These opening also serve as access to the controls and status screen on the GoPros. The mounting piece of the housing was designed to mate with standard GoPro screw connectors allowing it to be used on a variety of GoPro harnesses.

This bracket was mounted to the Action Mounts Chest mount. One bracket was mounted to the front of the chest mount and using the included GoPro mounting pads the chest harness was modified to carry a bracket on the back plate as well. The back plate of the Chest Harness was relatively small and when examining the footage taken during test runs the rear camera pair was found to be very unstable. The following diagram shows the chest harness.



Figure 4.4: Action Mounts chest mount showing the front mounting plate

The need to increase the stability without hindering wearability introduced a new specification to be incorporated into the design process. The part had to comfortably fit to the back of a runner while providing a larger surface area for the camera to mount to. The part shown in the following figure was created and tested to fulfil this specification.

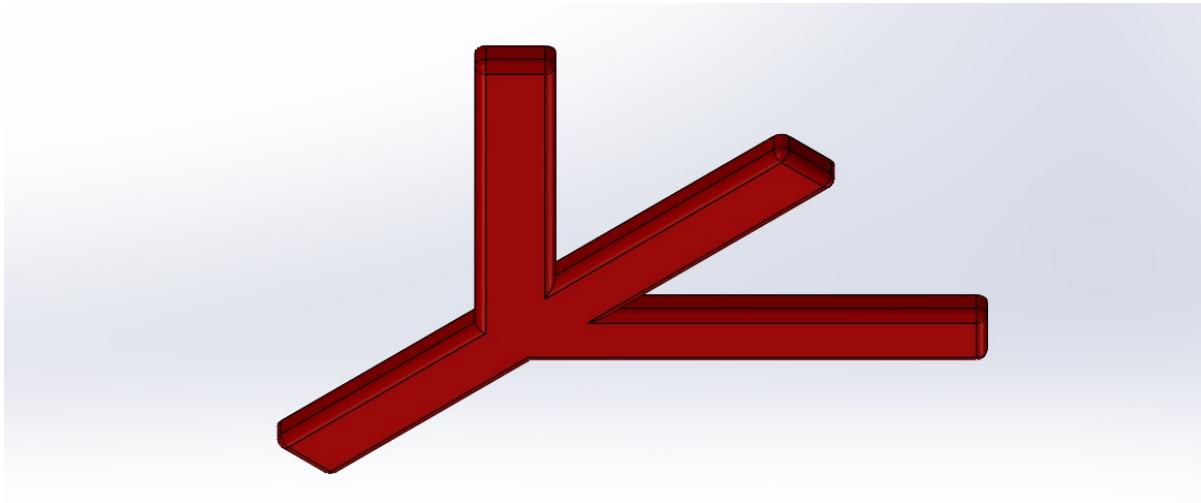


Figure 4.5: Stabilizer mounted to the rear camera

By 3D printing the above model and rounding the edges by hand the stabilizer was

created. The stabilizer was fastened to the back of the harness backplate. The following figure shows the stabilizer mounted below the rear camera, indicated with black arrows.

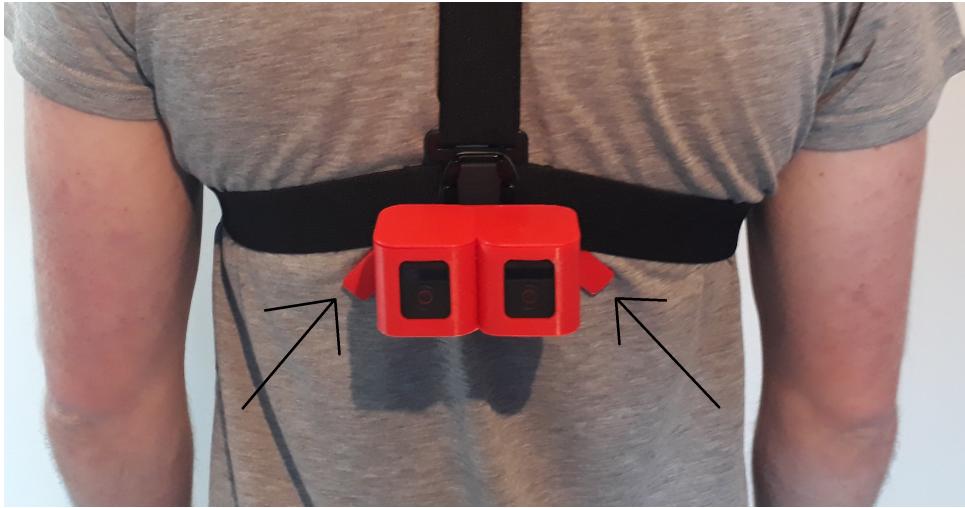


Figure 4.6: Stabilizer mounted to the rear camera

When comparing footage of the back cameras before and after the stabilizer was attached a clear difference is noted. The videos before contained excessive movement to the extent that the image blurred the lower limbs. This would prove problematic for attempts at image processing as the relatively low resolution and blurry images would introduce large uncertainties in the marker locations. The footage taken with the stabilizer was much smoother and less distorted, improving the visibility of the markers. After interviewing subjects about wearing the data capture system it was concluded that the stabilizer did not reduce the comfortability of the system.

### 4.3 Sony Xperia Z3 Compact Smartphone

The Sony Xperia Z3 Compact has a complex sensor system that includes an accelerometer, gyroscope, magnetometer, light sensor, pressure sensor, and proximity sensor. These sensors are combined locally to also create artificial sensors for orientation, gravity, and linear acceleration. A built in GPS is also available to provide global positional data. These sensors are updated constantly and pushed to different memory locations as explained in the Android API documentation.

To log the different datapoints generated by the sensors a software application for the smartphone was required. Many free applications are available on the Android Marketplace to fulfil this purpose; largely due to the open source approach the Android operating system is built on. By checking user reviews and ratings the following applications were considered.

### 4.3.1 AndroSensor

Androsensor was created by Fiv Asim and is available freely from the Android marketplace or [52]. The source code for this application is not available, but the author details the various underlying methods on [52]. The following are screenshots of the application running on the Z3 Compact.

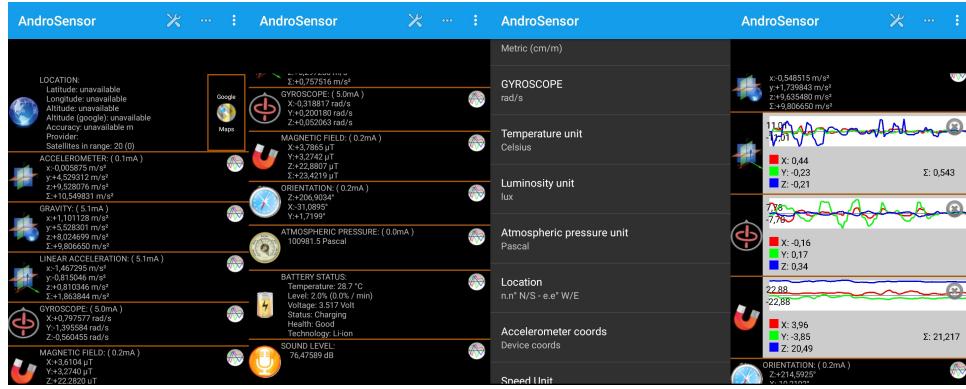


Figure 4.7: AndroSensor mobile application running on the Sony Xperia Z3 Compact

One of the advantages of AndroSensor is that it is highly configurable. A user can select which sensors are monitored and logged and the rate at which they are updated. This allows us to log important data and eases pre processing efforts. The application can also log the magnitude of sound allowing us to use this to synchronize the entire system. The output file type and units of the sensors can also be configured also easing pre processing.

There is a drawback related to not being able to view the source code of the application, we can not be sure if the data the application provides is raw real time data or filtered delayed data. This could introduce a delay to the system causing desynchronization of the different data sources.

### 4.3.2 SensorLab

Another popular application is SensorLab, created by LP Allis and available freely on the Google Play Store. This application is newer than AndroSensor and based on a more recent version of the Android APK. This should allow the application to run more efficiently and log data faster and more accurately. The application also provides real time sensor information in the form of various graphs and animations. The following are screenshots taken from within the SensorLab application.

This application has very little configurability. It automatically logs all sensor data at



Figure 4.8: SensorLab mobile application running on the Sony Xperia Z3 Compact

the maximum possible rate. After analysing the output file generated by SensorLab the logging rate was found to be close to 150Hz for most sensors. The parameters logged however did not include a volume reading, introducing synchronization difficulties.

The application is also closed source, making it impossible to know if the data values are processed or not. It was also unclear what units some of the data values were logged in, increasing the complexity of pre processing the data.

### 4.3.3 Custom Software

A custom software solution was developed by Stocks in [7] to log the various sensors of the smartphone. This application was developed to include readings from external sensors as well as the internal sensors of the smartphone. The source code for the application is available but is not well documented.

When the source code of the application was compiled using Android Studio the application did not function. The source code provided was not the final source code used for the research project. An attempt was made to modify the source code to correct the application but proved impossible due to the large codebase and lack of documentation. The following figure depicts screenshots from the 2 different versions of the application.



Figure 4.9: Screenshots of the custom software solution

#### 4.3.4 Final Selection

AndroSensor was selected over the other application due to the above mentioned advantages. The application was configured to log the accelerometer, gyroscope, magnetometer, GPS data, and sound level. The logging rate was set at 100Hz to have uniformity in the system. Once the data was logged the output file contained the following headings.

Heading	Fields
ACCELEROMETER	X,Y,Z
GRAVITY	X,Y,Z
LINEAR ACCELERATION	X,Y,Z
GYROSCOPE	X,Y,Z
MAGNETIC FIELD	X,Y,Z
ORIENTATION	X,Y,Z
ATMOSPHERIC PRESSURE	
LOCATION Latitude	
LOCATION Longitude	
LOCATION Speed	
LOCATION Orientation	
VOLUME	

Table 4.2: This table shows the different headings of the output AndroSensor file

## 4.4 Smartphone Mount Design

The design specification was to rigidly mount the smart phone to the chest of the subject. To complete this objective a rubber smartphone housing for the Z3 compact was fastened to the ActionMounts chest mount above the front duel camera. The following image illustrates the smartphone as connected to the harness. If the smartphone had been mounted on top of the camera we could have used pose estimation to understand exactly the orientation of the camera with respect to the world frame.



Figure 4.10: Smartphone mounted rigidly to the chest piece of the harness

## 4.5 Critical Point Markers

To increase the accuracy of the image processing elements of this methodology various colourful markers were used to identify critical points on the subjects lower extremities. The following picture shows the location of these markers.



Figure 4.11: Subject wearing Green and Pink Markers to identify critical points

From the image we can see that green markers were used to identify the toe edge of the runner in the front camera frame and pink markers to identify the end of the thigh in the same frame. In the rear mounted camera frame pink markers were used to identify the heel of the subject and green markers to identify the centre of the calf. These markers in conjunction with the model are used to identify the lower limb orientation during the fusion stage.

These luminous green and pink markers were chosen because they are bright in most lighting conditions and offer a stark contrast to both the subjects clothing and shoes. They also contrast a typical black tar road surface where the runs were performed. This contrast makes them easy to detect using feature detection as they cause a spike in the color decomposition of the image.

# Chapter 5

## Processing the Captured Data

This chapter is dedicated to the process of extracting critical data from the video files and the smartphone csv file. This process will take the raw captured data and transform it to values that we can feed into the Extended Kalman Filter. It is referred to in this work as preprocessing since the data is being processed before the filter.

### 5.1 Processing the IMU Data

The following flowcharts depicts the procedural processing of smartphone sensor and GPS data gathered from the chest mounted smartphone running the AndroSensor application. The following flow diagram shows the different steps in preprocessing.

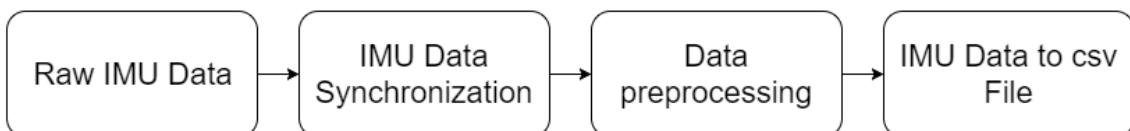


Figure 5.1: Diagram showing the progression and dependence of the major stages of sensor data processing in this project

From the diagram our raw input log file is a large unprocessed dataset captured by the smartphone with a logging rate of 100Hz. From this the critical data must be extracted and synchronized with the rest of the system. The data must also be manipulated to keep units constant and unify various frames of reference. By correctly preprocessing the data, the implementation of the EKF is simplified.

### 5.1.1 Obtaining IMU Data

Obtaining the raw sensor data was discussed in the previous chapter. The AndroSensor application was configured by specifying the units that all sensors must be logged in. The units of each sensor was selected as per the table below.

	Heading	Units
ACCELEROMETER	$m/s^2$	
GRAVITY	$m/s^2$	
LINEAR ACCELERATION	$m/s^2$	
GYROSCOPE	$rad/s$	
MAGNETIC FIELD	<i>microTesla</i>	
ATMOSPHERIC PRESSURE	<i>Pascal</i>	
LOCATION Latitude	<i>degrees</i>	
LOCATION Longitude	<i>degrees</i>	
LOCATION Speed	$m/s$	
LOCATION ORIENTATION	<i>degrees</i>	
VOLUME	<i>dB</i>	

Table 5.1: Table showing the different units of the output AndroSensor file

The importance of having uniform units is discussed later in this chapter. The above units have been chosen since they are SI units or the most relevant next possibility.

### 5.1.2 Synchronizing IMU Data

Previously the application used by [7] created a beep tone when the logging process was started. This beep tone was recorded by the cameras and was used as a common point of synchronization. Some drawbacks were identified by using this method. The smartphone would start logging only after the beep had completed. this meant that the smartphone data had to be resynchronized to the camera data by further testing. Another shortcoming was that the beep tone was not loud enough to comfortably be detected by the rear cameras, causing difficulty in noisy environments.

The AndroSensor application can record the the magnitude of sound that the microphone is experiencing at a given time. Using this magnitude we can see at what time instance a spike in volume (caused by a simple clap of the hands) was logged. This sample can be a common point between the IMU and the cameras and can thus be used to synchronize the different hardware elements.

Upon further consideration a point of synchronization should be created before and after the testing period to ensure that the different data streams stay in sync. This meant that

volume spike had to be made before and after each run.

### 5.1.3 Preprocessing IMU Data

All these variables have been recorded with respect the smartphone frame of reference. Since the smartphone was rigidly mounted to the body, a simple axis transformation could be completed to move the data onto the body frame of reference. This is further discussed in a following section.

Before we can input the sensor data directly to the EKF, we need to make some minor modifications to the data. This includes zeroing the positional data given by the GPS, since we assume our subject to start at position  $(0, 0)$  in the inertial frame. To zero the the following formulae was applied.

$$gpslat = (gpslat - gpslat_i) \times 110922$$

$$gpslong = (gpslong - gpslong_i) \times 92423$$

These algorithms subtracts the initial value of the GPS latitude and longitude from every value in the data vector and multiplies the difference by scalars to move the points to a NED frame of reference. These scalar values where obtained from [53].

$$gpsvelx = gpsspeed \times \cos(gpshead)$$

$$gpsvely = gpsspeed \times \sin(gpshead)$$

The above algorithms compute the x and y components from the GPS velocity vector by using the magnitude and heading logged from the sensor.

The accelerometer and gyroscope also contained some bias. This can be quantified by logging a large dataset while the smartphone is absolutely stationary and finding the average non-zero value of the individual sensors. Calculating the bias vectors we find them to be. For the accelerometer these values have been calculated as:

$$accelerometerX = -2.74252 \times 10^{-5} m/s^2$$

$$accelerometerY = -2.61902 \times 10^{-5} m/s^2$$

$$accelerometerZ = 2.04709 \times 10^{-5} m/s^2$$

The values for the gyroscope has been calculated as:

$$gyroscopeX = 5.46604 \times 10^{-5} r/s$$

$$gyroscopeY = 1.29292 \times 10^{-5} r/s$$

$$gyroscopeZ = 2.87219 \times 10^{-5} r/s$$

The logged data was modified to subtract these bias values.

### 5.1.4 Exporting IMU Data

The IMU data was finally exported as a final csv file and imported into MATLAB as a set of vectors. Each vector containing the samples of a specific data field. This allows for faster processing and access to the data.

The final step was to compute availability vectors for all the different data fields. This was done by creating a vector that contains a 1 if the sensor value has changed and a 0 if the value has not updated. Since many of the sensors on the smartphone do not update at 100Hz this was necessary to only input the values to the EKF if there was an update. When analysing these availability vectors we can see that the barometer only updates at a rate of 5Hz and the GPS at a rate of 1Hz.

## 5.2 Processing the Video Data

To further simplify the design and implementation of the EKF it is important to pre-process the video data to a less data heavy format than MP4. The following diagram shows the process of converting a large video file to a more lightweight csv file without losing any critical information.

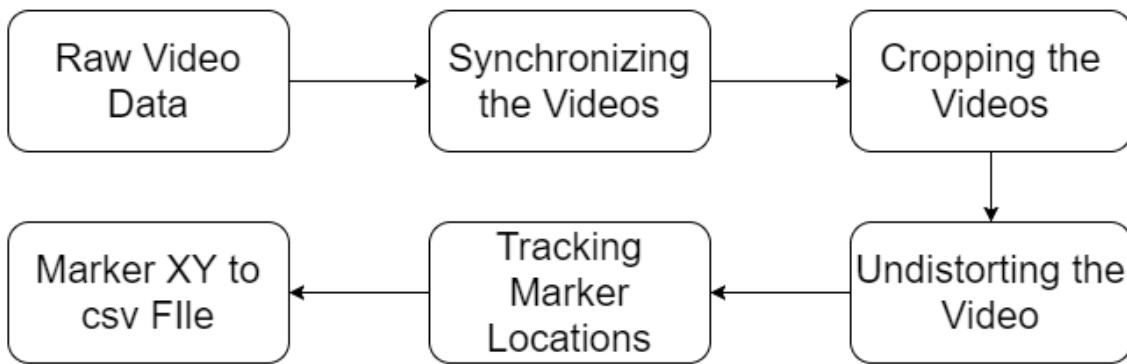


Figure 5.2: Diagram showing the progression and dependence of the major stages of video processing in the project

All raw video data streams must be synchronized and the important sections of the streams extracted. Correcting the various distortions introduced by the lens properties of the GoPro and decompressing the video files are necessary before image processing is attempted. Finally each marker position in every frame of all the individual cameras must be quantified. These pixel coordinates will serve as inputs to the EKF.

### 5.2.1 Obtaining Video Data

The cameras were housed in the custom designed dual camera housing shown in the previous chapter. Two cameras at the front capturing the motion of the lower limbs when they have a positive hip angle and two cameras mounted to the back to capture negative hip angles.

The GoPro cameras were all connected to a single GoPro remote and configured to record video when the remote was activated. After the harness had been properly secured the cameras were started. The remote starts the cameras at similar times, but not accurately enough to use as a method of synchronization. Below is a picture of the GoPro remote used.



Figure 5.3: GoPro camera remote control

### 5.2.2 Synchronizing Video Sources

I typical problem faced when working with different sources of data is that of synchronization. Since this project uses 4 different cameras, synchronizing the video sources are critical to generate accurate stereo vision and lower limb data.

The problem of synchronization was overcome by using a audio cue to align the video data post capturing. With all systems recording, a simple hand clap can serve as a spiking audio input easily identified in the audio track of the video streams. The frame associated with this audio spike can be identified using SVP (Sony Vegas Pro) video editing software as shown in the figure below.

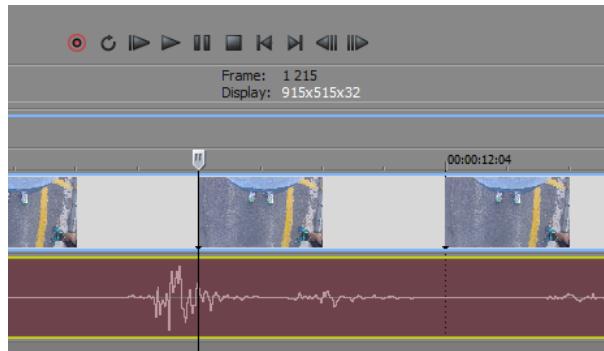


Figure 5.4: Figure showing the user interface of SVP video editing software

The red track in the above figure shows the recorded audio stream while the corresponding frames are displayed in the blue track above that. The cursor is aligned with the audio spike caused by the clap with the corresponding frame number displayed below the playback controls. This method was repeated for every video stream such that a common starting point was generated.

### 5.2.3 Cutting Critical Video Data

With the video data synchronized the next step was to generate a subset of video demonstrating a transient period and steady state period of running. By inspecting

the videos we can identify the different phases of the run and we can extract their frame numbers by once again using SVP.

A single video was used to determine the different points of the run. From synchronization to the end of the run was 18 seconds. This quantifies as 1800 frames as the cameras are recording at 100 frames per second.

A MATLAB script was created to cut the 1800 critical frames from the raw video file. This script also uncompressed the video files to a more data intensive .AVI file that contains a RGB vector for each individual pixel. The decompression of the video files makes them easier to manipulate as objects in MATLAB, but does however not restore any of the data lost in the process of compression. This MATLAB script can be found on the accompanying CD.

#### 5.2.4 Undistorting the Video Data

Due to the intrinsic lens properties of the GoPro cameras the edges of each frame had been severely distorted. To gain further understanding of undistorting video files [51] served as a reference. By using The built in camera calibration toolbox in MATLAB we can generate the various camera matrices needed to undistort the video files.

A custom MATLAB script was created to undistort the critical video segments. This script used the camera matrix in conjunction with build in functions to correct the lens distortion frame by frame of each video stream. The script can be found on the accompanying CD. The following image shows the difference before and after the frames were corrected.



Figure 5.5: Images demonstrating the effects of lens distortion.

### 5.2.5 Tracking and Exporting Marker Positions

This section discusses the different methods of image processing subdividing them into two main methodologies: automated and semi-automated. Each of these approaches offer advantages and disadvantages. To understand the approaches considered for this work it is important to visualize the input image data to the system. The following picture shows the various frames from all the cameras.



Figure 5.6: 4 Frames from the different cameras combined

The top row of images have been generated by the front mounted cameras and the bottom row by the rear mounted cameras. The left images were produced by the left cameras and the right images by the right cameras. From this image we can get a visual idea of the data our image processing system needs to manipulate.

Feature detection is a methodology in image processing that allows critical points of an image to be detected. This extracts an XY coordinate of the point of interest that can then be used to track the movement and position of an object in the image. This methodology can be applied to frames with more than one point of interest as shown in this work. Multiple points of interest in multiple video sources does however increase the complexity of feature detection.

An initial approach of using automated detection was chosen and three possible system were considered. Using a trained *neural network*, using an *edge detection* algorithm with a panning search algorithm and finally using a *colour identifying* algorithm paired with

a local search algorithm. These approaches all require a considerable amount of software to implement and therefore the viability of each has to be considered.

In theory a well trained *neural network* will provide the most robust and accurate system to identify features in varying lighting condition. It would also be the most accurate methodology for a system without the markers. This is due to the "understanding" developed by the neural network after sufficient training data is processed. Strong choices for neural network types would be a Hopfield network or a feed-forward network.

Training a network presents a key difficulty in setting up a neural network. The most important element of a neural network is the training data used to teach the network the correct node weightings. Training data consist of annotated images with descriptive metadata. Such a dataset needs to be created either automatically using image processing algorithm or can be created by computer assisted image processing. A neural network therefore cannot generate its own training data and can therefore only be used as a semi automated identification system.

The second approach considered was that of *edge detection*. This method identifies various edges in the image using discontinuities in brightness. Such a system can easily detect the edges of a marker, but cannot differentiate or classify edges. Pairing edge detection with a panning search algorithm allows us to search for marker shaped objects post edge detection. This algorithm needs to correctly identify the marker shapes and correctly differentiate and uniquely annotate the different markers as their respective points.

Although the implementation of edge detection is rather trivial, creating a search algorithm that can correctly identify changing marker shapes and assign them their correct point classification seems beyond the scope of this project. The following image shows edge detection as applied to a camera frame.

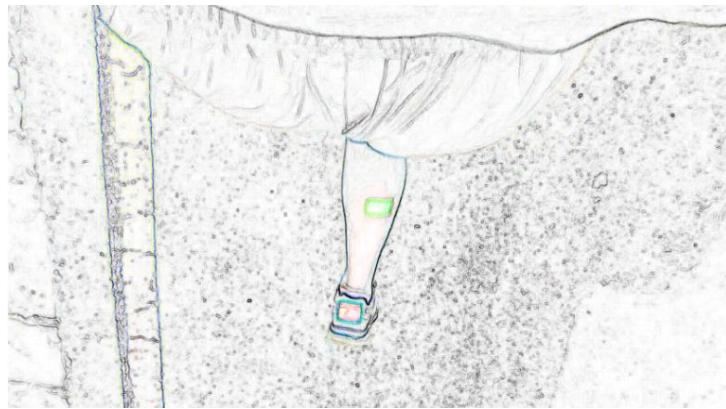


Figure 5.7: 4 Frames from the different cameras combined

From this example we can see that the marker edges are correctly identified, but the

colours do not correlate. As assumed the difficulty in this approach is not creating the image processing algorithm but rather in creating a panning search algorithm to extract marker information.

Another approach considered was that of a *colour identifying* algorithm. By using such methods the markers with their bright colours are easy to identify and differentiating between pink and green markers also seem trivial. The local search algorithm is also easy to implement and will start searching for marker in the next frame at the coordinates of the same marker identified in the previous frame.

Such an approach was partially implemented, but did not provide consistent enough results to use. These inconsistencies arise from the GoPro camera automatically adjusting its light level to best suit the environment. Changing the recorded light levels changes the perceived RGB values of the markers. This forces us to increase the colour rage of our markers, introducing more uncertainty and increasing the possibility false positives.

Another major drawback to using such a system is that it is not a generalized solution. This implies that the same markers must be used at the same positions every time the experiment is run. It also depends heavily on the markers, stunting the idea of transitioning to a markerless system.

By comparing the proposed automatic methods above and considering ease of implementation and accuracy no single method seems viable. This is either due to the lack of example data (in the case of the neural network), or the limitations of computer algorithms (too complex in the case of edge detection and not general purpose or accurate enough in the case of the colour identifier).

The final approach considered and used was to *semi automatically* label critical point in the image using a toolbox created by Hedrick et al. [54]. This software allows for a semi automatic tracking of points of interest in the video frames by using user input in conjunction with computer algorithms. While this method is labour intensive it is arguably more accurate than the previously investigated methodologies and more generalized. The following figure shows the functionality of the toolbox within MATLAB.

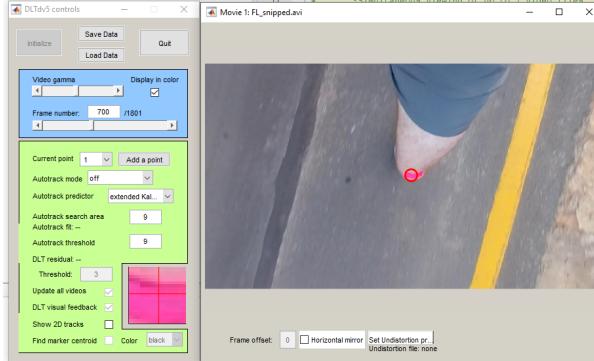


Figure 5.8: Using the dltdv toolbox in MATLAB to export marker coordinates

By using this toolbox we can generate a dataset of  $(x, y)$  pixel coordinates by identifying the different markers on each frame and assigning their corresponding points to them. The following tables shows the markers and their corresponding point.

Point	Identifier	Marker Colour
<b>Front Cameras</b>		
Point 1	Right Knee	PINK
Point 2	Left Knee	PINK
Point 3	Right Foot	GREEN
Point 4	Left Foot	GREEN
<b>Rear Cameras</b>		
Point 1	Right Calf	GREEN
Point 2	Left Calf	GREEN
Point 3	Right Heel	PINK
Point 4	Left Heel	PINK

Table 5.2: Table showing the relation between points and markers

Each video stream was processed and a total of 16 different coordinate sets was generated, 8 from the front cameras and 8 from the rear cameras. These coordinates were output as csv files that could easily be imported into MATLAB.

Finally availability vectors were created for each of the 16 points. From these vectors we can also calculate the percentages of each camera seeing each marker. This can give us an idea as to how often the cameras are contributing data and how much of the total gait is captured. These percentages are discussed in a following chapter.

## On units

In order to assure consistency between the different sources of data considering a general set of units is of critical importance. It was decided to implement the system using SI units such that all lengths were given in meters and all angles in radians.

## On Frames of Reference

We need to develop some commonalities among the various sensors and their individual frames of reference. This thesis considers 4 different frames of reference, namely: the body frame, the smartphone frame, the camera frame and the inertial frame.

The body frame has been defined as shown in the image below.



Figure 5.9: Figure demonstrating the body frame of reference

The smartphone frame is shown in this image below.



Figure 5.10: Figure demonstrating the frame of reference of the smartphone

Since the smartphone is rigidly mounted to the body a simple axis transformation is required to transform the sensor data to the body frame. The following equations show the transformation.

$$bodyX = smartphoneZ$$

$$bodyY = -smartphoneY$$

$$bodyZ = -smartphoneX$$

There is however another frame of reference that is important to understand, that of the camera frame.

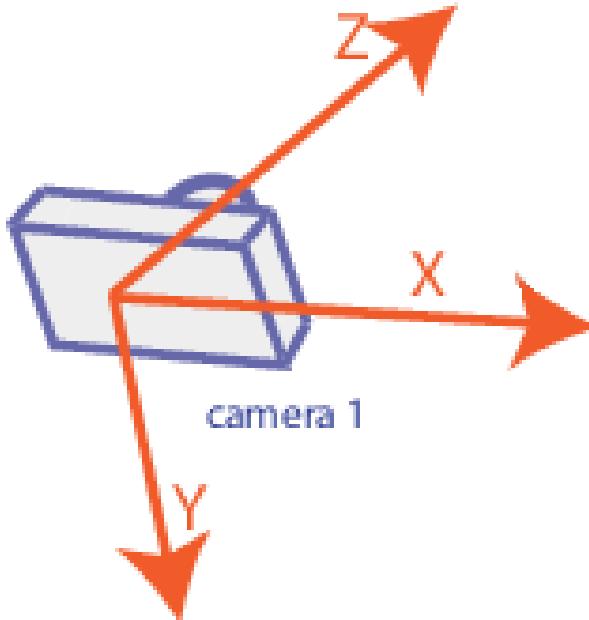


Figure 5.11: Figure demonstrating the camera frame of reference, image adapted from [12]

Since the cameras are pointing down when the subject is standing still the Z axis of the camera aligns with the Z axis of the body. The X and Y axis are transformed during fusion.

# Chapter 6

## Data Fusion and State Estimation

This chapter is dedicated to explaining the mathematical methods and models used to fuse data generated by the cameras and smartphone. It details the design and implementation of an EKF using MATLAB software. Kim et al. [55] and source code provided by Patel served as the foundation of this chapter. All files pertaining to the filter have been included on the accompanying CD.

A good starting point for this chapter would be to define the states of interest of our system. We choose our states to be various parameters relating to the human gait and the bodies pose and position in the inertial frame. The model used in this work has various constants (the lengths of the rigid beams making up the limbs), but many changing parameters (the angles these rigid beams make with each other). These angles quantify the position of the various joints during steady state running and have therefore been chosen as states.

We cannot directly measure these angles since we have no sensors directly connected to these points. All the system states are predicted by the EKF using various measurements relating directly and indirectly to the system states. The following table specifies the parameters used to symbolize the states of the system.

State	Description
$x_{body}$	x Position of body w.r.t. the inertial frame
$y_{body}$	y Position of body w.r.t. the inertial frame
$z_{body}$	z Position of body w.r.t. the inertial frame
$\phi_{body}$	Roll of body w.r.t. the inertial frame
$\theta_{body}$	Pitch of body w.r.t. the inertial frame
$\psi_{body}$	Yaw of body w.r.t. the inertial frame
$\theta_{LH}$	Pitch of left thigh w.r.t. left hip
$\psi_{LH}$	Yaw of left thigh w.r.t. left hip
$\theta_{LK}$	Pitch of left calf w.r.t. left knee
$\theta_{LA}$	Pitch of left foot w.r.t. left ankle
$\theta_{RH}$	Pitch of right thigh w.r.t. right hip
$\psi_{RH}$	Yaw of right thigh w.r.t. right hip
$\theta_{RK}$	Pitch of the right calf w.r.t. right knee
$\theta_{RA}$	Pitch of the right foot w.r.t. the right ankle

Table 6.1: Table showing the different states of the model to be determined by the EKF

Our system is not only concerned with these positional and angular elements, but also how they change over time. These rates where defined as the derivative of the states with respect to time. The first derivative serving as velocity and angular velocity, while the second derivative serves as acceleration and angular acceleration. Here the vector  $\mathbf{x}$  serves as a element of our state vector  $\mathbf{X}$ .

$$\mathbf{x} = [x_{body} \ y_{body} \ z_{body} \ \phi_{body} \ \theta_{body} \ \psi_{body} \ \theta_{LH} \ \psi_{LH} \ \theta_{LK} \ \theta_{LA} \ \theta_{RH} \ \psi_{RH} \ \theta_{RK} \ \theta_{RA}]$$

$$\mathbf{X} = [\mathbf{x} \ \dot{\mathbf{x}} \ \ddot{\mathbf{x}}]$$

The vector  $\mathbf{x}$  contains 14 elements. From the above equations it is clear that our state vector  $\mathbf{X}$  contains 42 elements. With the states of our system defined we must discuss state estimation and the critical elements of the EKF.

## 6.1 State Estimation

The key purpose of the Kalman Filter is the ability to estimate the various states of our system. To understand how this estimation will occur we need to firstly understand the KF. The following diagram shows the critical parts of the KF.

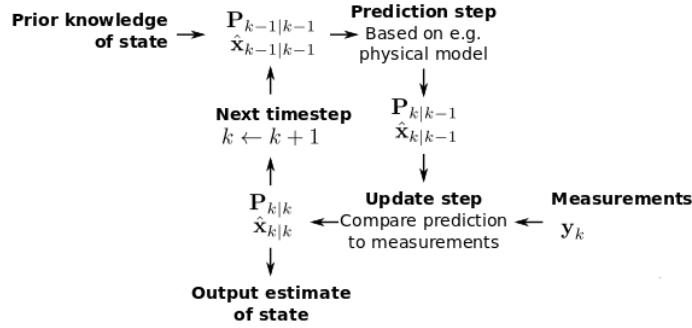


Figure 6.1: Figure showing the interplay of various elements of the KF, adapted from [56]

The critical elements of the KF, as seen in the diagram, are: the states themselves, a prediction model, an update model, measurements, initialization values, and various uncertainty models.

The underlying equation of the KF is the *process equations*, where the state value at the next time instance  $X_{k+1}$  is predicted by applying the transition matrix  $F_{k+1}$  to the state values at the current time instance  $X_k$  and adding a zero mean Gaussian noise term  $w_k$ . The  $w$  element of this equation is collected in the process noise matrix  $Q$ .

$$X_{k+1} = F_{k+1}X_k + w_k$$

Another underlying equation of the KF is the *measurement equation*. Here the observable measurements contained in  $Y_k$  are related to the states at the current time instance  $X_k$  through the measurement matrix  $H_k$ . A zero mean Gaussian noise term  $v_k$  is added to account for measurement uncertainty. The various measurement uncertainties are collected in the measurements noise matrix  $R$ .

$$Y_k = H_kX_k + v_k$$

As discussed in the literature review, the KF can only be applied to linear systems. The EKF is the extension of the filter so that it may be applied to nonlinear systems. To introduce the workings of the EKF we start by defining a classic nonlinear system defined by the state space model

$$X_{k+1} = f(k, X_k) + w_k$$

$$Y_k = h(k, X_k) + v_k$$

In this representation  $w_k$  and  $v_k$  are consistent with definitions given for the KF and are contained in matrices  $Q$  and  $R$  respectively. From the state space equations one can see that we now predict states using a nonlinear transition matrix function  $f$  and  $h$ .

These functions can be defined by their anti-derivative matrices  $F$  and  $H$  respectively.

$$F_{k+1,k} = \frac{\delta f(k, X_k)}{\delta x}$$

$$H_k = \frac{\delta h(k, X_k)}{\delta x}$$

Now that we have defined the fundamental equations of the KF and EKF we can discuss in more detail the implementations of the various components required by the filter.

## 6.2 Prediction Equations

The fundamental assumption made when deriving the prediction equations for our states was that the acceleration (both linear and angular) was constant between sampling intervals. It would therefore stand that the positional states of the filter  $[x_{body} \ y_{body} \ z_{body}]$  and the angular states of the filter  $[\theta_{body} \dots \theta_{RA}]$  could be predicted using:

$$\begin{aligned}\ddot{p}_{k+1} &= \ddot{p}_k + \sigma_p^2 \\ \dot{p}_{k+1} &= \dot{p}_k + \ddot{p}_k T + \sigma_p^2 \\ p_{k+1} &= p_k + \dot{p}_k T + \sigma_p^2\end{aligned}$$

for the positional states, and:

$$\begin{aligned}\ddot{\alpha}_{k+1} &= \ddot{\alpha}_k + \sigma_\alpha^2 \\ \dot{\alpha}_{k+1} &= \dot{\alpha}_k + \ddot{\alpha}_k T + \sigma_\alpha^2 \\ \alpha_{k+1} &= \alpha_k + \dot{\alpha}_k T + \sigma_\alpha^2\end{aligned}$$

for the angular states.

These prediction equations were created in MATLAB using symbolic functions. The

sigma associated each equations takes into account the prediction uncertainties contained in the Q matrix. By adjusting these values we can increase the filter performance.

## 6.3 Measurement Equations - Body

These equations relate the various states relating to the body with respect to the inertial frame. Since the measurements from the smartphone are related to this frame of reference the measurement equations were very simple.

The accelerometer gave us the linear acceleration of the body and therefore the measurement, transformed by rotational matrix gave us the body states with respect to the inertial frame. The gyroscope also transformed by a rotational matrix gives us the rotational states of the body. The magnetometer can be used to estimate the pose of the body states by solving the rotational parameters. The barometer value can be converted to a height value of the body with respect to the surface of the earth. Finally the GPS data gives us the absolute positional location of the body with respect to the inertial frame.

## 6.4 Measurement Equations - Lower Limbs

The measurement equations for the lower limbs where generated using forward kinematics.

### 6.4.1 Euler Matrices

The following matrices are the rotational matrices for rotating a point in 3D space along a certain axis. These matrices are used in conjunction with forward kinematics to solve for different points along the rigid beam model.

$$Roll(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{bmatrix}$$

$$Pitch(\theta) = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix}$$

$$Yaw(\psi) = \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

### 6.4.2 Forward Kinematics

By using forward kinematics we can determine the position of each marker by applying our states as elements of the rotational matrices.

In the **front** camera frame we have:

Knee Markers

$$KneeXYZ = bodyY + bodyZ + R1^T \times Thigh$$

$$R1 = Pitch(Hip) \times Yaw(Hip)$$

Toe Markers

$$ToeXYZ = bodyY + bodyZ + R1^T \times Thigh + R2^T \times Calf + R3^T \times Foot$$

$$R2 = Pitch(Knee)$$

$$R3 = Pitch(Ankle)$$

In the **rear** camera frame we have:

Calf Midpoints Markers

$$CalfXYZ = bodyY + bodyZ + R1^T \times Thigh + R2^T \times 0.5 \times Calf$$

$$R1 = Pitch(Hip) \times Yaw(Hip)$$

$$R2 = \text{Pitch}(Knee)$$

Heel Markers

$$\text{HeelXYZ} = \text{bodyY} + \text{bodyZ} + R1^T \times \text{Thigh} + R2^T \text{Calf}$$

## 6.5 Camera Matrix

It is important to understand the different parameters that mathematically quantify cameras. These parameters can be divided into *intrinsic* and *extrinsic* camera matrices. Extrinsic camera variables related to the cameras position in the inertial frame and the direction the camera is facing. These can be summarized by the extrinsic camera matrix:

$$[R | t] = \left[ \begin{array}{ccc|c} r_{1,1} & r_{1,2} & r_{1,3} & t_1 \\ r_{2,1} & r_{2,2} & r_{2,3} & t_2 \\ r_{3,1} & r_{3,2} & r_{3,3} & t_3 \end{array} \right]$$

The intrinsic camera matrix describes the geometric properties of the camera.

$$K = \begin{pmatrix} f_x & s & x_0 \\ 0 & f_y & y_0 \\ 0 & 0 & 1 \end{pmatrix}$$

The XYZ points generated by the forward kinematics are then transferred from the body frame to the camera frame by using the camera matrix. This takes us from our initial angular state values to expected pixel coordinates.

## 6.6 Q Matrix, R Matrix and Initialization

This section will discuss the final components of the EKF namely the Q matrix containing the various process noise variations, R matrix containing the various measurement noise variances and the initial state values.

### 6.6.1 Defining the Q Matrix

From the above equations we can see that the Q matrix must have dimensions of  $n * n$ , where  $n$  in this equation is the total amount of states. As previously defined in this section our filter operates over 42 states, giving Q a size of  $42 * 42$ . All the variance parameters will be contained on the diagonal of the matrix with all other entries being zero.

To find the initial values of these uncertainties the derivative of the various accelerations were taken. The maximum element from that set was selected as the uncertainty parameter.

### 6.6.2 Defining the R Matrix

The R matrix relates to the measurement variables and must therefore have size of  $m * m$ , where  $m$  is the amount of inputs the EKF. These are elements of the sensors themselves and can be found by researching the relative data sheets for the smartphone IMU. As for the cameras a relatively large uncertainty of about 5 pixels was assumed.

### 6.6.3 Choosing Initial States

The subject was stationary during the initial stages of the run. This allows us to initialize our state vector with all states initially zero. The filter will therefore track the transience and steady state estimation of a running subject, giving insight into the filters performance under different conditions.

### 6.6.4 Initializing the Covariance Matrix P

Following on from the previous section zeroing the initial states of the system allows us to have a relatively small initial covariance matrix due to the relative certainty we have that the states are truly zero.

## 6.7 MATLAB Implementation

All the above equations and variables have been implemented in MATLAB and has been included on the accompanying CD. The prediction and measurement equations were implemented in MATLAB and then exported to individual functions to be used in the KF. The Jacobians of these functions were also generated by using the symbolic math toolbox within MATLAB.

# Chapter 7

## Results and Discussion

### 7.1 Results

The project had various critical phases that were completed. This section will present all the results generated relating to the major concepts of the project. This includes results relating to: system design, image processing, captured and processed data, and the EKF. All results are generated from an 18 second run completed with the designed system. This run generated 1800 data points, including a transient period from samples 0 to 200. Samples 400 to 1800 demonstrates steady state running.

#### 7.1.1 System Design Results

The data capture system was designed to meet the specifications outlined in the methodology chapter. The device was comfortable, light weight and adjustable such that it could easily be used by a large variety of people. The device was also sturdy and well constructed allowing for long term usage. Pictured below is a subject wearing the data capture system.



Figure 7.1: subject wearing the motion capture system

The device in no way affected the gait of the subject as the system had a minimal footprint. The stereo camera housings were fully functional and offered a good fit for both the cameras and the mounting hardware. Although the cameras moved during running they captured all the critical points in every single frame. This can be seen in the scatter plots presented in Appendix B.

### 7.1.2 Image Processing Results

The video streams were successfully processed to extract all the critical data points from each frame. By using the DigitizingTools MATLAB package created by Hedrick et al. [54] it was possible to accurately quantify the marker positions on the different video frames.

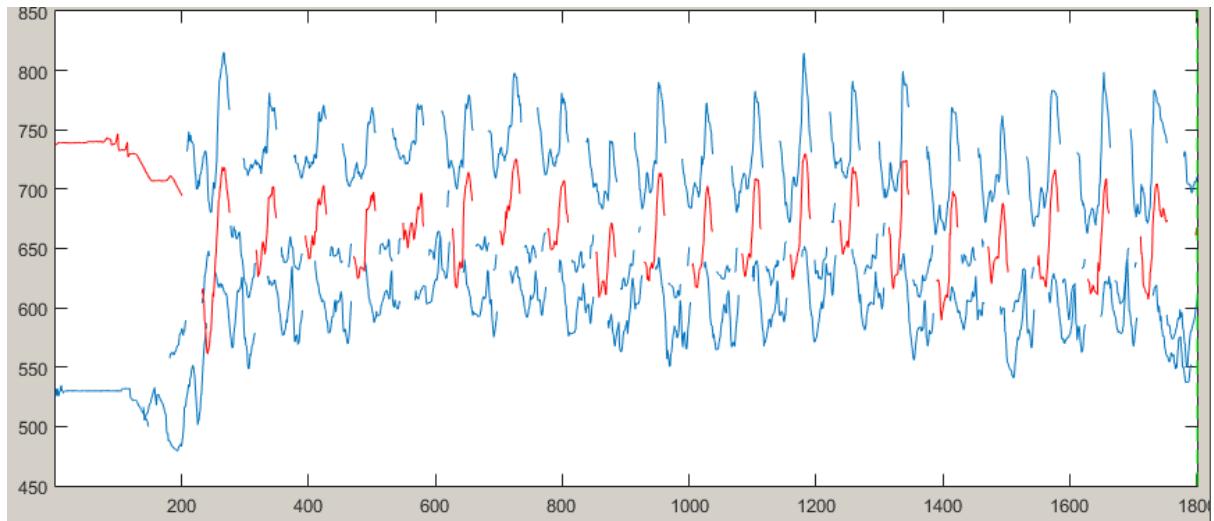


Figure 7.2: iamge processing things

The figure above shows the quantization of various points identified in the image frames of a single video stream. The frame number is displayed on the x-axis while the y-axis shows tracks the change in position of the  $(x, y)$  pixel values. Four points were tracked in each video stream. since there were 4 cameras each tracking 4 points a total of 16 datasets were successfully created. From the scatter plots presented in appendix B we can see that the markers did not exit the frame of the camera.

### 7.1.3 Captured and Processed Data

The following figure shows the pre EKF accelerometer data in the body frame. The x-axis represents the sample and the y-axis linear acceleration in  $m/s^2$ .

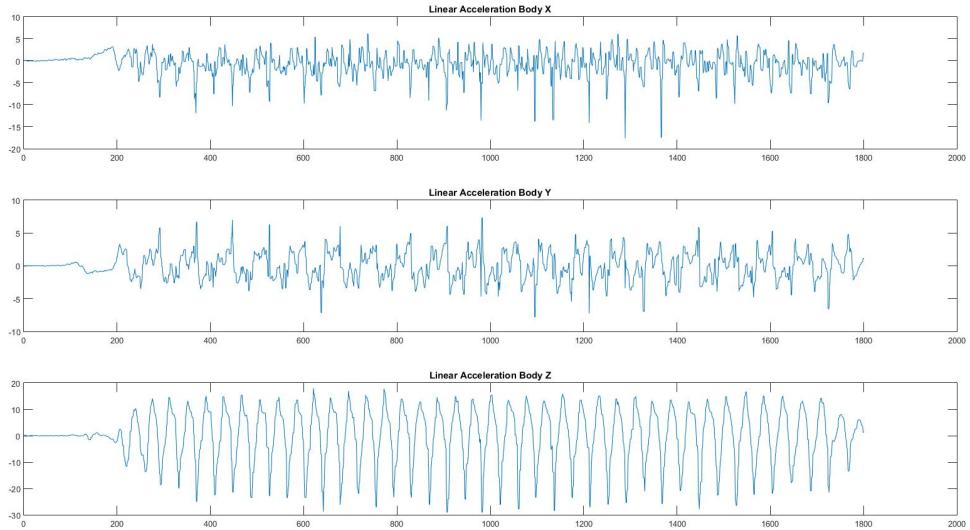


Figure 7.3: Pre filter accelerometer data

The accelerometer data has been pre processed to remove the sensor bias. When looking at the Z axis accelerometer data we can clearly see a periodic motion correlating with the gait period of the subject. Some clear points of local minimum indicate when the runner made contact with the ground. These values were in the expected range of 3 to 4 times the gravitational acceleration. The different local maxima around  $10m/s^2$  indicated the runner reaching his highest point during the gait, where he is not in contact with the ground and experiencing only gravitational acceleration.

After testing it was determined that the smartphone accelerometer has a range of  $80m/s^2$  functioning between  $-40m/s^2$  and  $40m/s^2$ . It is thus clear that the readings on all the accelerometer axis stayed within these bounds.

The following figure shows the pre EKF gyroscope data in the body frame. The x-axis represents the sample and the y-axis angular velocity in  $rad/s$ .

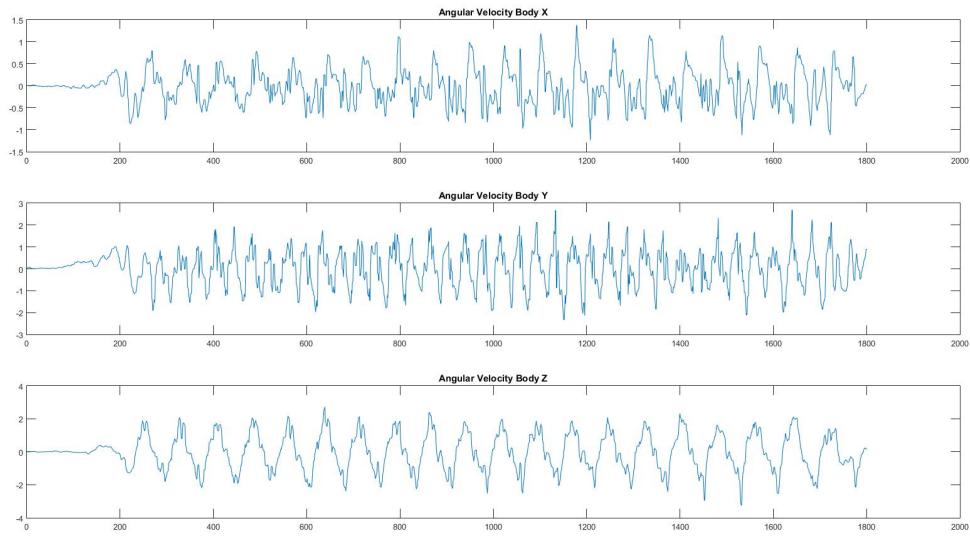


Figure 7.4: Pre filter gyroscope data

The bias has been removed from the above gyroscope data. By looking at the Z axis we can clearly see a periodic swinging of the chest. This chest rotation is a natural gait element to balance the angular moment generated by the hips.

All the readings were comfortably within the bounds of the gyroscope that has a range of  $60rad/s$  allowing readings from  $-30rad/s$  to  $30rad/s$ .

The x-axis of the GPS Position graph represents position in the x direction and the y-axis the position in the y direction of the body with respect to the inertial frame. The y-axis of the GPS Velocity graph represents velocity and the x-axis represents the sample number. Velocity in the x direction is shown in blue and velocity in the y direction is shown as red.

The barometer graph plots the pressure on the y-axis against the sample number on the x-axis. The barometer is shown to be quite sensitive picking up the vertical motion of the runner as changes in pressure. A clear pattern is visible directly correlating with the gait period of the runner. The above details are shown in the figure below.

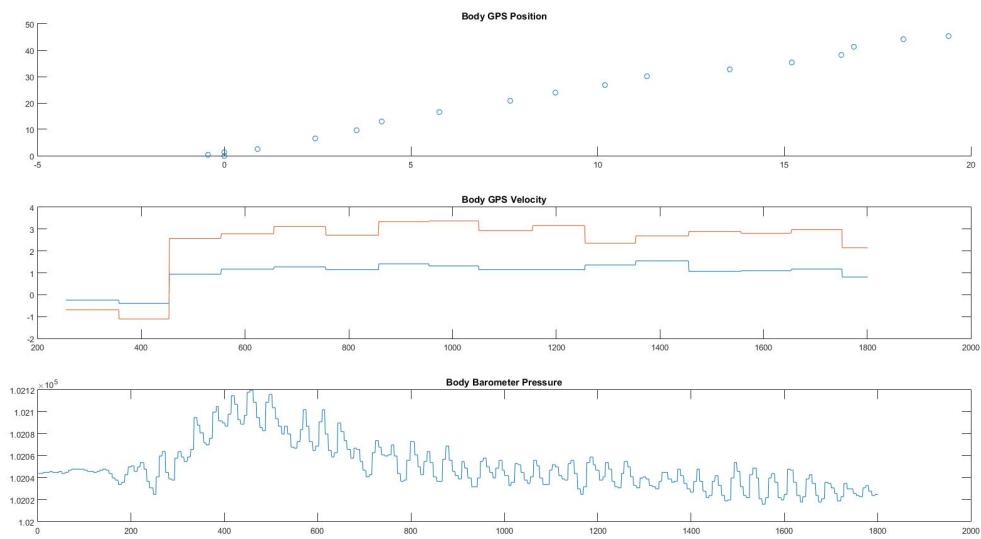


Figure 7.5: Pre filter GPS and barometer data

Since this thesis is to serve as a proof of concept, it was important to quantify how often the system actually captured valuable data. The following figure shows how often a measurement is available as a percentage. This includes how often each camera sees the different markers and how often the smartphone sensors are available.

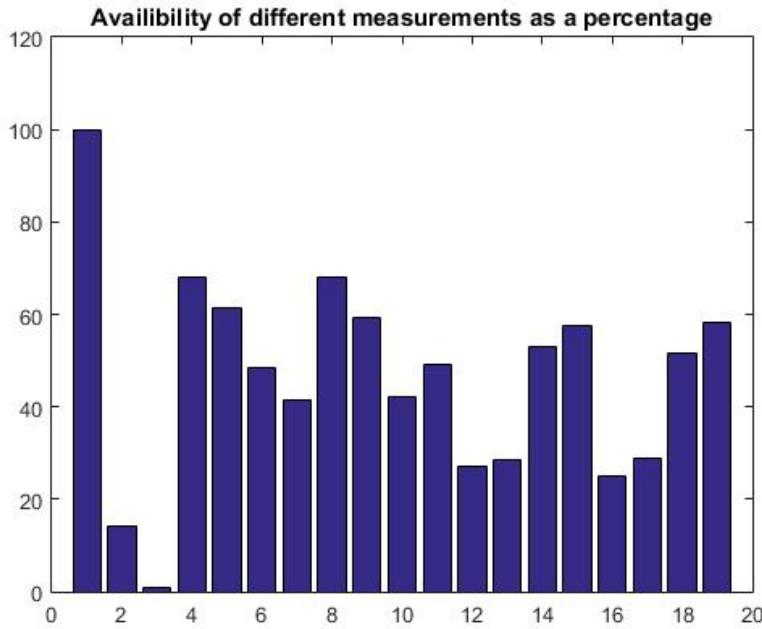


Figure 7.6: Bar graph showing the availability of various measurements as a percentage

The y-axis of this bar chart represents the percentage that the marker is available during the test, while the x-axis are the different measurements. Another set of figures included in appendix B shows in more depth the availability of the various data sources.

#### 7.1.4 EFK Results

3 Different configurations of the final EKF was run and the results are presented in appendix B. The initial version of the filter was unmodified and with all tuning parameters as derived in previous chapters. The results of this filter were unstable, with the base states growing exponentially.

The second version of the filter had reduced the prediction uncertainties and increased the measurement uncertainties, this version of the filter was more stable but did still not produce realistic values for the 14 base states. From this we can conclude that the measurement equations were generating larger errors than expected.

The final version of the filter decreased the measurement uncertainties and increased

the values process uncertainties. This filter performed the worst and the states grew exponentially at the fastest rate. From this we can confirm that our assumption of constant acceleration does not accurately predict the motion of the system and requires more concrete measurements to unify.

The filter itself functions, but all attempted configurations produced divergent estimates. This can be rectified by tuning the filter more accurately or adapting the measurement equations to take into account the camera pose so that the camera frame and the body frame can be unified more accurately. This will be discussed further in the following section.

## 7.2 Discussion

### 7.2.1 System Design

The system was successfully designed and constructed using available equipment, and 3D printable parts. The system is functional and can be used for future experiments.

### 7.2.2 Image Processing

The techniques of image processing considered in this thesis shows the difficulty in implementing an automated image processing system. For this system the use of a semi-automatic image processing solution was the most efficient and viable.

### 7.2.3 Captured Data

The captured data provided insight into various elements of the problem space as a whole. The gyroscope readings conclusively showed the oscillating motion of the chest that was not included in the model. The accelerometer readings gave us a method of quantifying the gait period by measuring the samples between foot impacts. The sensor data captured from the smartphone proved accurate and the ease of makes smartphones viable tools as modern sensors.

Furthermore the availability vectors showed that as a proof of concept the system is often capturing data that can be quantified as various gait parameters. This shows the importance of having both front and rear mounted cameras as well as the ability of these

cameras to capture critical lower limb data. With the high availability percentages and the gait being captured both in the forward and rear frame the system captures enough data to quantify the gait, given a method of fusion.

### 7.2.4 System Modelling

By using the proposed rigid beam model it was possible to correctly implement measurement equations quantifying the lower limb states. To this extent, the model was successfully. The model was also born from existing literature and has been shown to be viable.

The system model had 2 major shortcomings. Firstly the assumption of a rigid chest that stays aligned with the hips and secondly the assumption that the cameras stay aligned with the body frame.

It is clear from the Gyroscope data that the chest of a runner oscillates. Since the cameras protrude from the chest they move in a related arc pattern in space. This was not taken into account by the model and introduced large errors when projecting marker point from the model in the body frame to the image in the camera frame.

The camera frame also changes relative to the body frame. This can be rectified by more accurately determining the pose of the camera and incorporating such parameters into the forward kinematics of the model. This will however require a sensor directly mounted to the camera to determine the pose, or a well constructed model of how the camera moves during a run.

The assumption were made to ease the initial implementation of the work. Since this project is novel the first step was to test a system and find shortcomings. These shortcomings have clearly been identified and can be rectified in the second implementation of this system.

### 7.2.5 Extended Kalman Filter

The filter was fully implemented in MATLAB. The final results produced by the filter were divergent and could therefore not correctly estimate the lower limb angles. The previous section explains shortcoming relating to the model of the system that caused erroneous variables as inputs to the EKF.

The filter itself however robust and modular and the various parameters and models of the filter can easily be changed to incorporate various model limitations. From this it

may be possible to get positive results from the filter after slight modification.

Unfortunately no truth data was generated during this research to compare to the data generated from the wearable motion capture system. This could have improved the various uncertainty models of the EKF.

# Chapter 8

## Conclusions and Future Work

This chapter is dedicated to drawing conclusions based on results found and make recommendation on future iterations of this project. Since the underlying methodology is quite novel this system should serve as the foundational step to a fully automated motion capture system.

### 8.1 Conclusion

The aim of this project was threefold: design a novel wearable data capture system using cameras and sensors to record various parameters of a running bipedal subject, develop a kinematic model to interpret the gathered data points, and design an estimation algorithm to process the captured data. These goals were achieved within certain constraints as discussed in the results section.

Due to the relatively small dataset used in this project it serves as a proof of concept and a foundational work that can be iteratively improved. This system serves as an initial design and understanding the shortcomings are the most critical elements of such a project.

All milestones were successfully implemented and provided mostly positive results bar the results produced by the EKF. This was due to the model shortcomings and camera pose assumptions made as discussed in the previous chapter. These limitations could be eliminated by modelling the chest as a rotating oscillator. The camera pose can be estimated from this new model or sensors can be used to estimate the camera pose.

## 8.2 Future work

The system was originally designed with four cameras due to the availability of equipment and the assumption that stereo vision would be implemented. For this a set of front and rear mounted cameras would be necessary. Originally there was an intention to reduce the total amount of cameras.

The iterative design process would have used the following roadmap. Initially using 4 cameras and an smartphone as a sensor. Then reducing the system to two cameras (one mounted to the back of the runner and one mounted to the chest of the runner) and a smartphone as a sensor. The next iteration would use the the smartphone camera at the front and a single camera at the back whilst using the smartphone as a sensor as well. This system greatly reduces the cost of the original design philosophy, even given that a powerful and expensive modern smartphone would be needed. The final iteration would use only the smartphone at the front as a single camera and sensor. For this method to work the estimation algorithm would need to better understand the periodic motion of the human gait and the model would need to increase in complexity.

There is however a fundamental shortcoming in the proposed methodologies. As discussed the uncertainty in the camera pose creates difficulties when trying to align the different frames of reference. This can be rectified by using purpose built IMUs attached to the cameras to better estimate their pose and incorporate it into the kinematic equations.

Due to the labour intensive approach taken to image processing a further avenue for improvement is the automation of feature detection. The progression would start from the current implementation of markers and the use of a semi-automatic toolbox to identify critical points on the image. Next, computer algorithms should be created to automate the image processing with markers. This could allow for much longer runs and larger datasets to be studied, introducing elements such as fatigue and other running modifiers. The next iterative step would be to remove the markers from the runner such that the set up time of the system is reduced. This is a difficult problem to solve using classical image processing as the variables relating to the runner and the environment are not constant. Perhaps a neural network can be trained to identify the different elements of the lower limbs.

These improvements would decrease the overall cost of the system and optimize the the process substantially. This decrease in hardware does imply that the complexity of the underlying algorithms and models would increase. This trade-off can be considered for future developments.

Finally the results obtained have not been verified against a true data source. An experiment to verify these results was designed, where external cameras would generate three dimensional positional data of the markers. This can be compared to the data generated from the EKF as applied to the wearable motion capture system by creating visual animations from both sources and comparing them. The system can also be verified by using a full body sensor based motion capture suit as designed by XSens.

# Bibliography

- [1] M. Sandau, H. Koblauch, T. B. Moeslund, H. Aanæs, T. Alkjær, and E. B. Simonsen, “Markerless motion capture can provide reliable 3d gait kinematics in the sagittal and frontal plane,” *Medical engineering & physics*, vol. 36, no. 9, pp. 1168–1175, 2014.
- [2] A. Pfister, A. M. West, S. Bronner, and J. A. Noah, “Comparative abilities of microsoft kinect and vicon 3d motion capture for gait analysis,” *Journal of medical engineering & technology*, vol. 38, no. 5, pp. 274–280, 2014.
- [3] D. Roetenberg, H. Luinge, and P. Slycke, “Xsens mvn: full 6dof human motion tracking using miniature inertial sensors,” *Xsens Motion Technologies BV, Tech. Rep*, 2009.
- [4] T. Seel, J. Raisch, and T. Schauer, “Imu-based joint angle measurement for gait analysis,” *Sensors*, vol. 14, no. 4, pp. 6891–6909, 2014.
- [5] VICON, “Vicon motion capture system,” <https://www.vicon.com/motion-capture/biomechanics-and-sport>, [Online; accessed 16-October-2017].
- [6] A. Patel, B. Stocks, C. Fisher, F. Nicolls, and E. Boje, “Tracking the cheetah tail using animal-borne cameras, gps, and an imu,” *IEEE Sensors Letters*, vol. 1, no. 4, pp. 1–4, 2017.
- [7] B. Stocks, “Cheetah motion analysis,” Master’s thesis, University of Cape Town, South Africa, 2016.
- [8] A. Patel and M. Braae, “Rapid acceleration and braking: Inspirations from the cheetah’s tail,” in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE, 2014, pp. 793–799.
- [9] J. M. Benyus, “Biomimicry: Innovation inspired by nature.”
- [10] L. Chen, H. Wei, and J. Ferryman, “A survey of human motion analysis using depth imagery,” *Pattern Recognition Letters*, vol. 34, no. 15, pp. 1995–2006, 2013.

- [11] P. Picerno, “25 years of lower limb joint kinematics by using inertial and magnetic sensors: A review of methodological approaches,” *Gait & posture*, vol. 51, pp. 239–246, 2017.
- [12] MATHWORKS, “Matlab,” <https://www.mathworks.com/products/matlab.html>, [Online; accessed 10-October-2017].
- [13] DASSAULTSYSTEMES, “Solidworks,” <http://www.solidworks.com/>, [Online; accessed 10-October-2017].
- [14] K. Kaneko, F. Kanehiro, S. Kajita, K. Yokoyama, K. Akachi, T. Kawasaki, S. Ota, and T. Isozumi, “Design of prototype humanoid robotics platform for hrp,” in *Intelligent Robots and Systems, 2002. IEEE/RSJ International Conference on*, vol. 3. IEEE, 2002, pp. 2431–2436.
- [15] K. Kaneko, S. Kajita, F. Kanehiro, K. Yokoi, K. Fujiwara, H. Hirukawa, T. Kawasaki, M. Hirata, and T. Isozumi, “Design of advanced leg module for humanoid robotics project of meti,” in *Robotics and Automation, 2002. Proceedings. ICRA ’02. IEEE International Conference on*, vol. 1. IEEE, 2002, pp. 38–45.
- [16] BOSTONDYNAMICS, “Atlas, the next generation,” <https://www.youtube.com/watch?v=rVlhMGQgDkY>, [Online; accessed 19-October-2017].
- [17] AGILITYROBOTICS, “Cassie the bipedal ostrich bot,” <https://www.youtube.com/watch?v=5YDIzeW42hg>, [Online; accessed 1-November-2017].
- [18] BOSTONDYNAMICS, “Atlas,” <https://www.bostondynamics.com/atlas>, [Online; accessed 1-November-2017].
- [19] AGILITYROBOTICS, “Cassie,” <http://www.agilityrobotics.com/robots/>, [Online; accessed 2-November-2017].
- [20] K. E. Adolph and S. R. Robinson, “The road to walking: What learning to walk tells us about development,” *Oxford handbook of developmental psychology*, vol. 1, pp. 403–443, 2013.
- [21] S. Hanson and A. Jones, “Is there evidence that walking groups have health benefits? a systematic review and meta-analysis,” *Br J Sports Med*, vol. 49, no. 11, pp. 710–715, 2015.
- [22] K. R. Fox, “The influence of physical activity on mental well-being,” *Public health nutrition*, vol. 2, no. 3a, pp. 411–418, 1999.

- [23] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs,” *arXiv preprint arXiv:1606.00915*, 2016.
- [24] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, “Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 1874–1883.
- [25] TESLA, “Tesla autopilot,” <https://www.tesla.com/autopilot>, [Online; accessed 6-October-2017].
- [26] R. H. Taylor, A. Menciassi, G. Fichtinger, P. Fiorini, and P. Dario, “Medical robotics and computer-integrated surgery,” in *Springer handbook of robotics*. Springer, 2016, pp. 1657–1684.
- [27] S. A. Kane and M. Zamani, “Falcons pursue prey using visual motion cues: new perspectives from animal-borne cameras,” *Journal of Experimental Biology*, vol. 217, no. 2, pp. 225–234, 2014.
- [28] H. C. Pearson, P. W. Jones, M. Srinivasan, D. Lundquist, C. J. Pearson, K. A. Stockin, and G. E. Machovsky-Capuska, “Testing and deployment of c-viss (cetacean-borne video camera and integrated sensor system) on wild dolphins,” *Marine Biology*, vol. 164, no. 3, p. 42, 2017.
- [29] M. Gabel, R. Gilad-Bachrach, E. Renshaw, and A. Schuster, “Full body gait analysis with kinect,” in *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*. IEEE, 2012, pp. 1964–1967.
- [30] E. E. Stone and M. Skubic, “Unobtrusive, continuous, in-home gait measurement using the microsoft kinect,” *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 10, pp. 2925–2932, 2013.
- [31] R. A. Clark, K. J. Bower, B. F. Mentiplay, K. Paterson, and Y.-H. Pua, “Concurrent validity of the microsoft kinect for assessment of spatiotemporal gait variables,” *Journal of biomechanics*, vol. 46, no. 15, pp. 2722–2725, 2013.
- [32] V. Gikas and H. Perakis, “Rigorous performance evaluation of smartphone gnss imu sensors for its applications,” *Sensors*, vol. 16, no. 8, p. 1240, 2016.
- [33] Google, “Sensors overview,” [https://developer.android.com/guide/topics/sensors/sensors\\_overview.html](https://developer.android.com/guide/topics/sensors/sensors_overview.html), [Online; accessed 11-October-2017].

- [34] Y. Wang, L. Wang, T. Yang, X. Li, X. Zang, M. Zhu, K. Wang, D. Wu, and H. Zhu, “Wearable and highly sensitive graphene strain sensors for human motion monitoring,” *Advanced Functional Materials*, vol. 24, no. 29, pp. 4666–4670, 2014.
- [35] L. Cai, L. Song, P. Luan, Q. Zhang, N. Zhang, Q. Gao, D. Zhao, X. Zhang, M. Tu, F. Yang *et al.*, “Super-stretchable, transparent carbon nanotube-based capacitive strain sensors for human motion detection,” *Scientific reports*, vol. 3, p. 3048, 2013.
- [36] M. Shoaib, S. Bosch, O. D. Incel, H. Scholten, and P. J. Havinga, “Complex human activity recognition using smartphone and wrist-worn motion sensors,” *Sensors*, vol. 16, no. 4, p. 426, 2016.
- [37] N. Sun and Y. Sakai, “New approaches to human gait simulation using motion sensors,” in *Advanced Information Networking and Applications Workshops (WAINA), 2017 31st International Conference on.* IEEE, 2017, pp. 126–131.
- [38] F. E. Zajac and J. M. Winters, “Modeling musculoskeletal movement systems: joint and body segmental dynamics, musculoskeletal actuation, and neuromuscular control,” *Multiple muscle systems: Biomechanics and movement organization*, vol. 8, pp. 121–148, 1990.
- [39] F. E. Zajac, R. R. Neptune, and S. A. Kautz, “Biomechanics and muscle coordination of human walking: Part i: Introduction to concepts, power transfer, dynamics and simulations,” *Gait & posture*, vol. 16, no. 3, pp. 215–232, 2002.
- [40] ——, “Biomechanics and muscle coordination of human walking: part ii: lessons from dynamical simulations and clinical implications,” *Gait & posture*, vol. 17, no. 1, pp. 1–17, 2003.
- [41] J. Soechting and M. Flanders, “Moving in three-dimensional space: frames of reference, vectors, and coordinate systems,” *Annual review of neuroscience*, vol. 15, no. 1, pp. 167–191, 1992.
- [42] A. Patel and M. Braae, “Rapid turning at high-speed: Inspirations from the cheetah’s tail,” in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on.* IEEE, 2013, pp. 5506–5511.
- [43] BOSTONDYNAMICS, “Robots,” <https://spectrum.ieee.org/automaton/robotics/humanoids/next-generation-of-boston-dynamics-atlas-robot>, [Online; accessed 11-October-2017].
- [44] Q. Chen, R.-H. Fu, and Y.-C. Xu, “Electrical circuit analogy for heat transfer analysis and optimization in heat exchanger networks,” *Applied Energy*, vol. 139, pp. 81–92, 2015.

- [45] D. C. Karnopp, D. L. Margolis, and R. C. Rosenberg, *System dynamics: modeling, simulation, and control of mechatronic systems*. John Wiley & Sons, 2012.
- [46] GOPRO, “Gopro fetch (dog harness),” <https://shop.gopro.com/EMEA/mounts/fetch-dog-harness/ADOGM-001.html>, [Online; accessed 22-October-2017].
- [47] ——, “Hero session,” <https://shop.gopro.com/EMEA/cameras/hero-session/CHDHS-104-master.html>, [Online; accessed 10-October-2017].
- [48] SONY, “Xperia z3 compact,” <https://www.sonymobile.com/za/products/phones/xperia-z3-compact/>, [Online; accessed 10-October-2017].
- [49] ACTIONMOUNTS, “Chest mount,” <http://action-mount.com/products/chest-mount/>, [Online; accessed 10-October-2017].
- [50] M. Christensen, “Gopro hero4 session,” <https://grabcad.com/library/gopro-hero4-session-1>, [Online; accessed 6-October-2017].
- [51] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, ISBN: 0521540518, 2004.
- [52] F. Asim, “Androsensor,” <https://play.google.com/store/apps/details?id=com.fivasim.androsensor&hl=en>, [Online; accessed 10-October-2017].
- [53] “Maritime safety information,” <http://msi.nga.mil/MSISiteContent/StaticFiles/Calculators/degree.html>, [Online; accessed 10-November-2017].
- [54] T. L. Hedrick, “Software techniques for two-and three-dimensional kinematic measurements of biological and biomimetic systems,” *Bioinspiration & biomimetics*, vol. 3, no. 3, p. 034001, 2008.
- [55] P. Kim, *Kalman filter for beginners: with MATLAB examples*. CreateSpace, 2011.
- [56] WIKIMEDIA, “Kalman filter diagram,” [https://commons.wikimedia.org/wiki/File:Basic\\_concept\\_of\\_Kalman\\_filtering.svg](https://commons.wikimedia.org/wiki/File:Basic_concept_of_Kalman_filtering.svg), [Online; accessed 2-November-2017].

## Appendix A: Calibration

---

### Vision Calibration

Before calibrating the stereo cameras and their singular quantifications it is important to take a step back and determine the camera matrix for each individual camera. This allows us to model the cameras mathematically individually. This will allow us to accurately model elements such as lens distortion, principal ray offset and uncertainties accurately. The pinhole camera model is used to quantify camera parameters.

MATLAB has a built in stereo camera calibration application that can calibrate a set of cameras and create an object containing all the essential parameters, including the individual camera intrinsics and relative camera extrinsic.

To calibrate the cameras pictures of a white and black chequerboard is fed in to the calibration application that then mathematically determines the camera parameters. In order to make this process more efficient a video of the moving chequerboard was taken and various frames of importance extracted using a simple MATLAB script. The code for this script and the calibrated camera models can be found on the accompanying CD

### Smartphone Calibration

The smartphone sensor is calibrated during the construction of the smartphone. The hardware detail of the sensor could not be found as the sensor model is not identified. The only correction in smartphone sensor data was to detect the various sensor biases. This was explored in chapter 6 of the report.

## Appendix B: Results Figures

---

The following is a list of the graphs generated by MATLAB

1. IMU, Barometer, and GPS availability Graph
2. Front Left Camera availability Graph
3. Front Right Camera availability Graph
4. Back Left Camera availability Graph
5. Back Right Camera availability Graph
6. Front Left and Front Right Scatter Plots
7. Back Left and Back Right Scatter Plots
8. EKF Configuration 1 - Actual State Values
9. EKF Configuration 1 - Estimated State Values
10. EKF Configuration 2 - Actual State Values
11. EKF Configuration 2 - Estimated State Values
12. EKF Configuration 3 - Actual State Values
13. EKF Configuration 3 - Estimated State Values

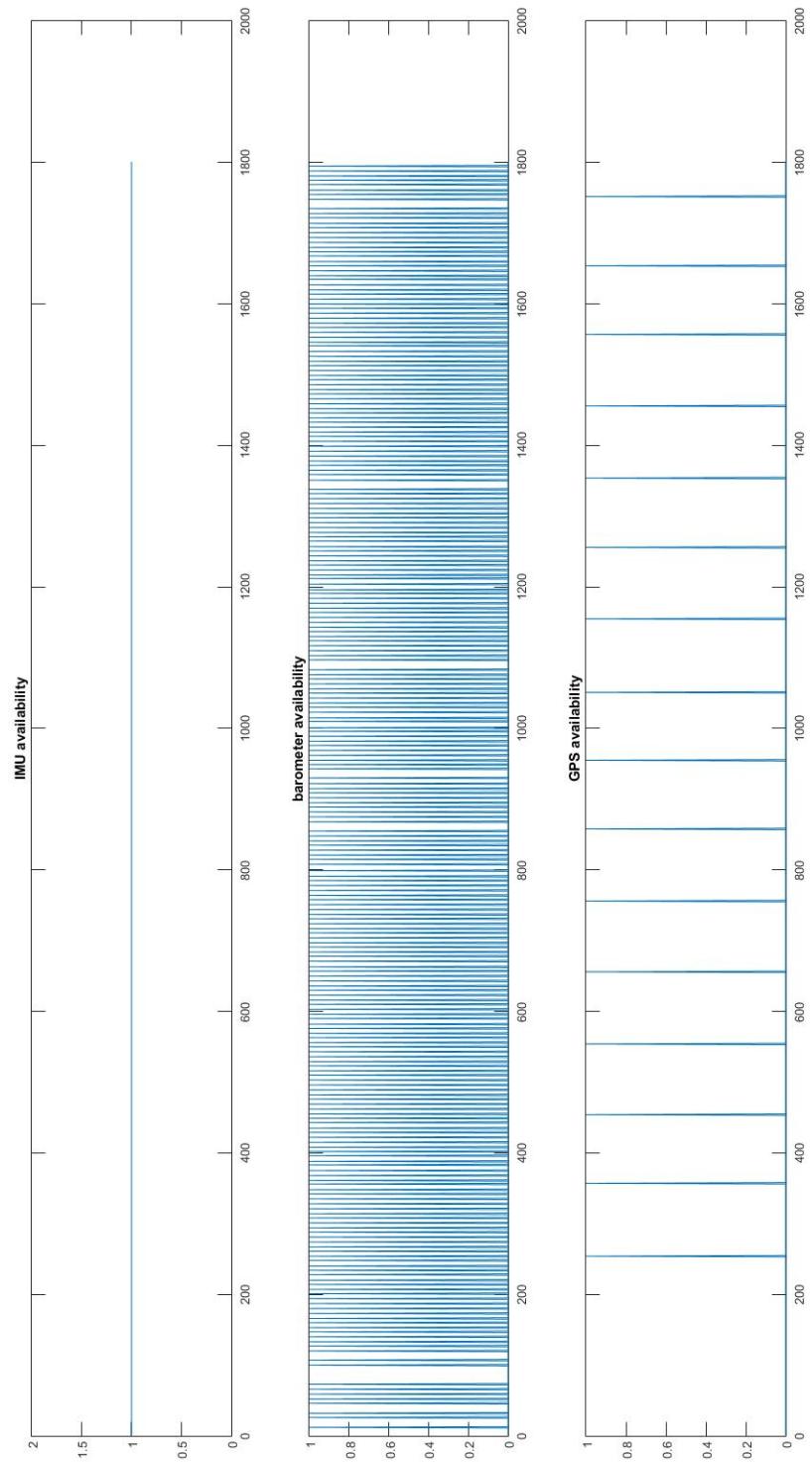


Figure 8.1: IMU, Barometer, and GPS availability

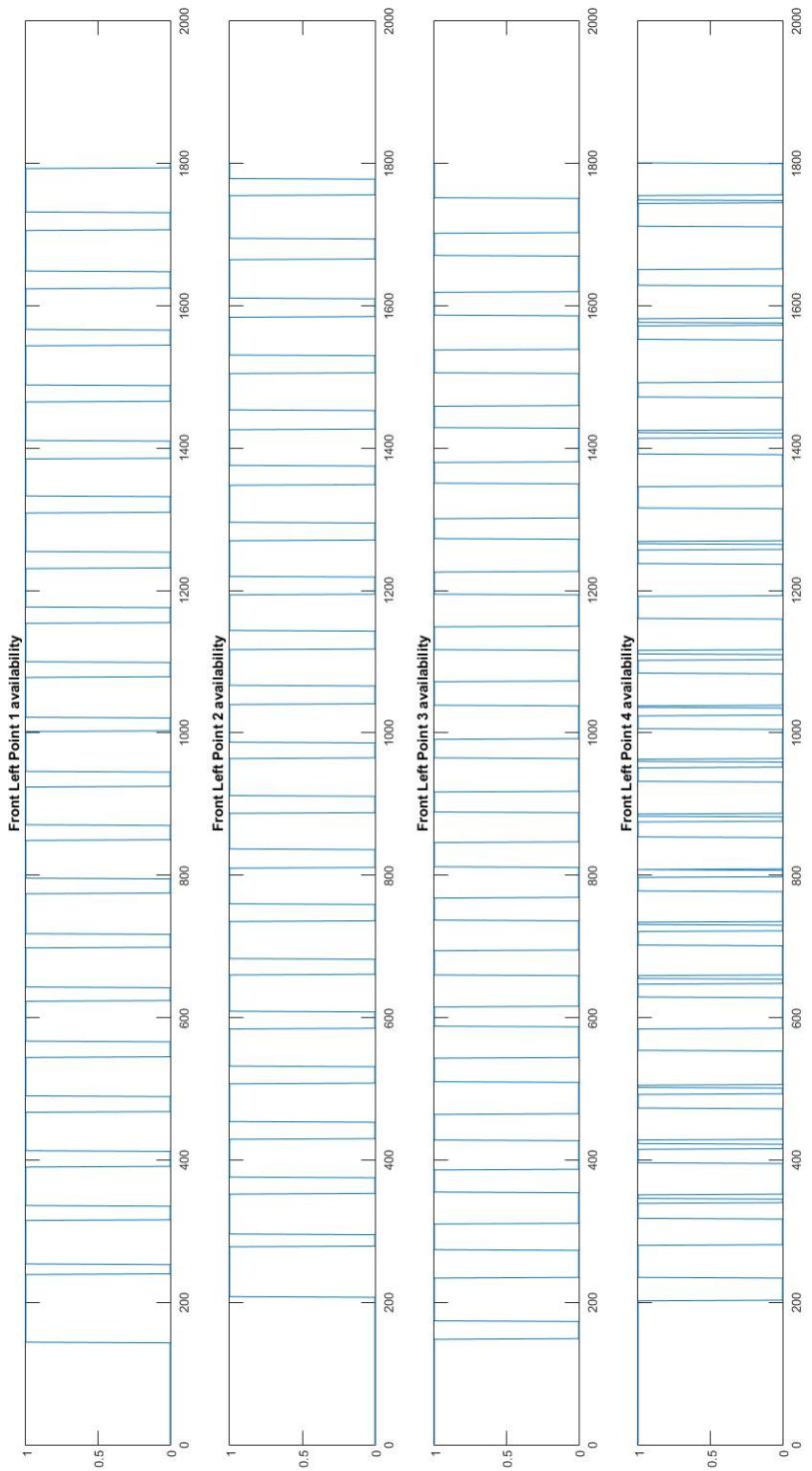


Figure 8.2: Front Left Camera availability

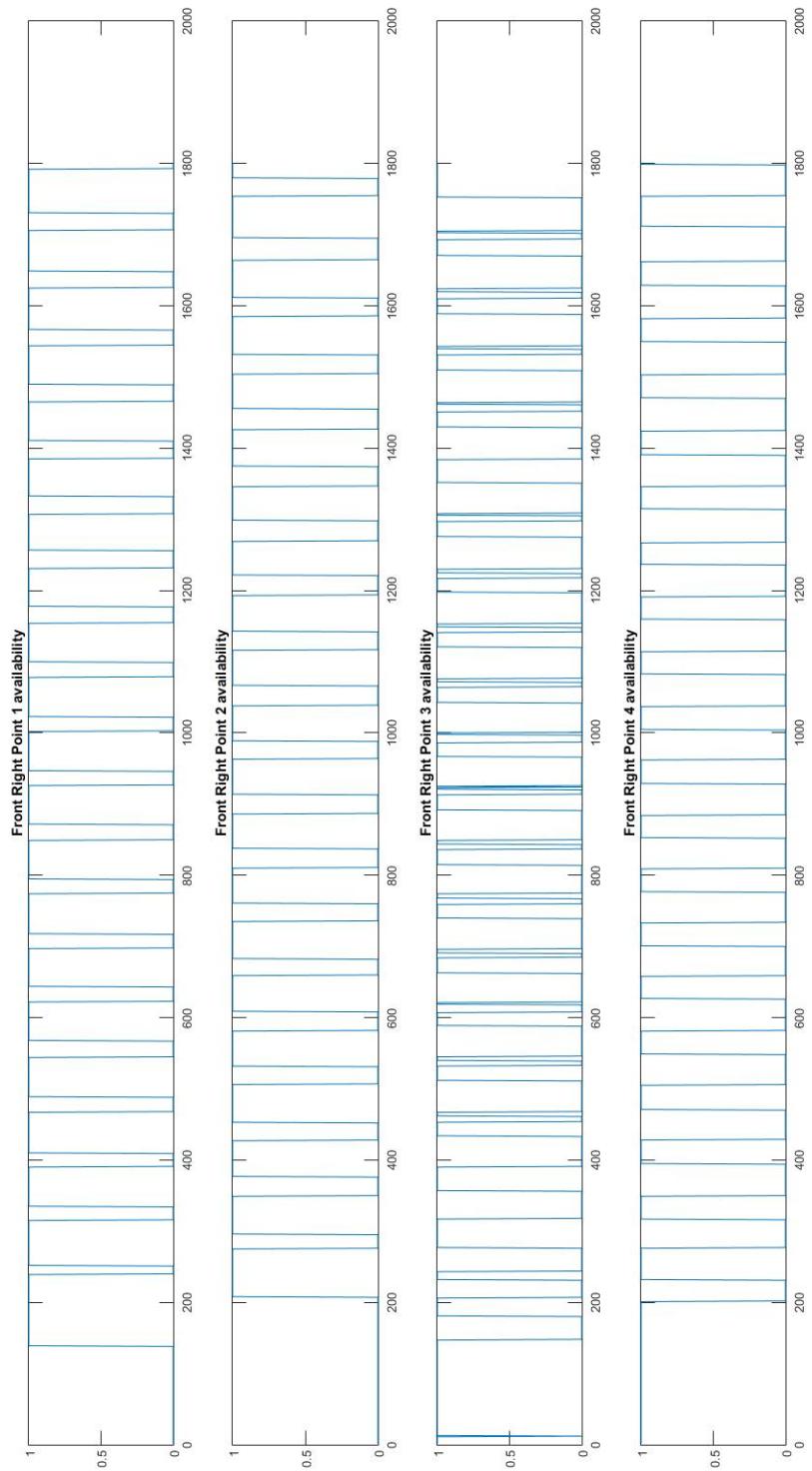


Figure 8.3: Front Right Camera availability

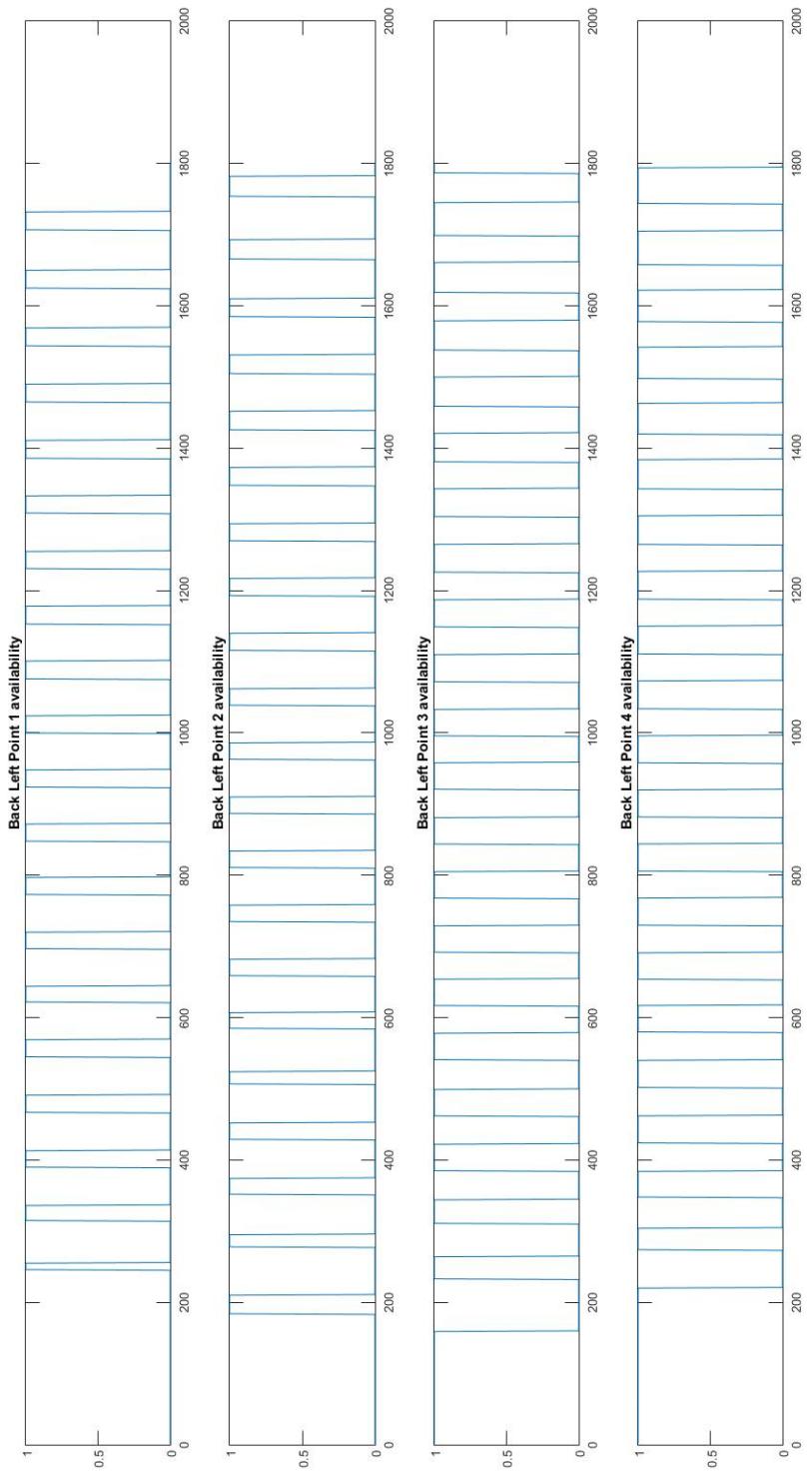


Figure 8.4: Back Left Camera availability

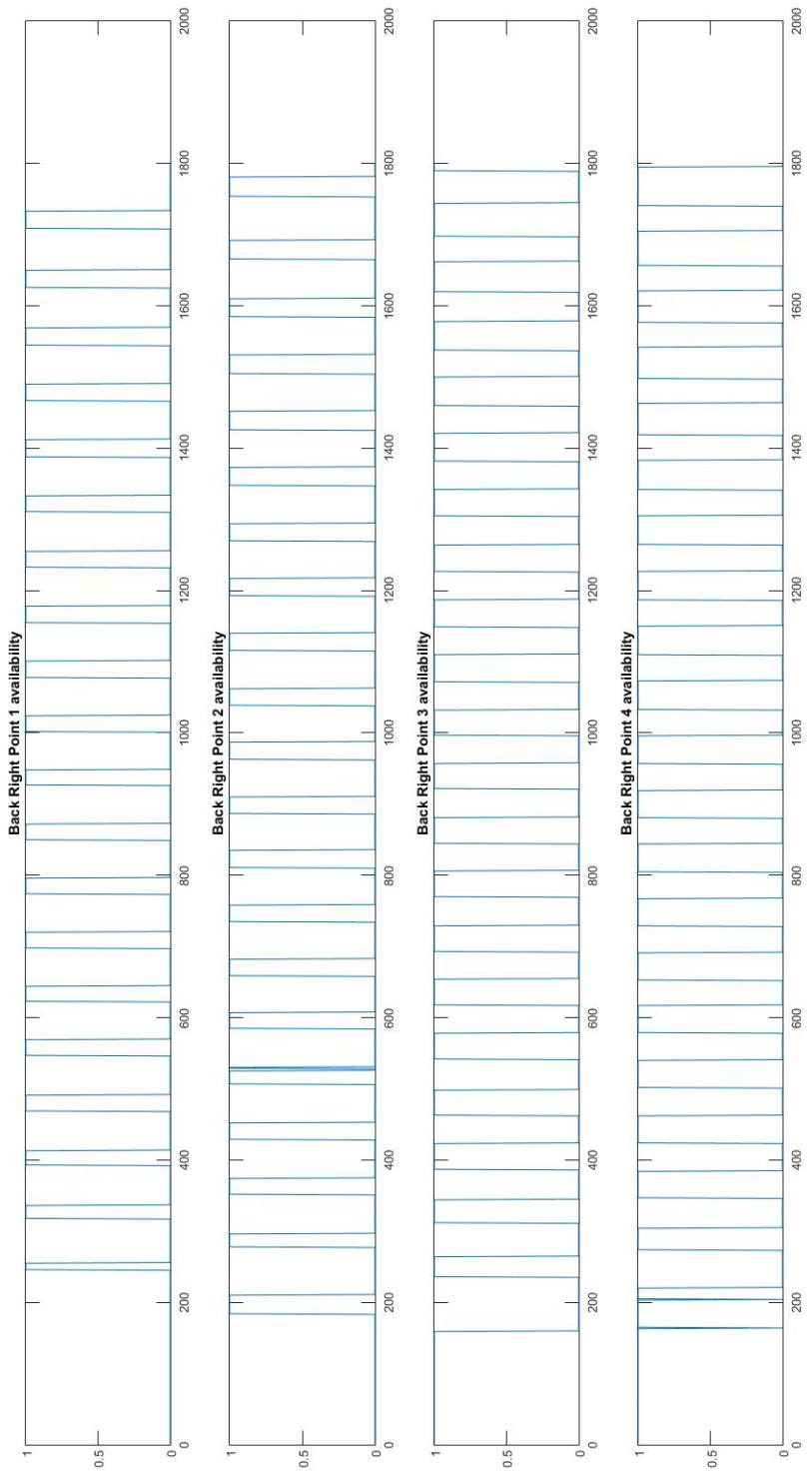


Figure 8.5: Back Right Camera availability

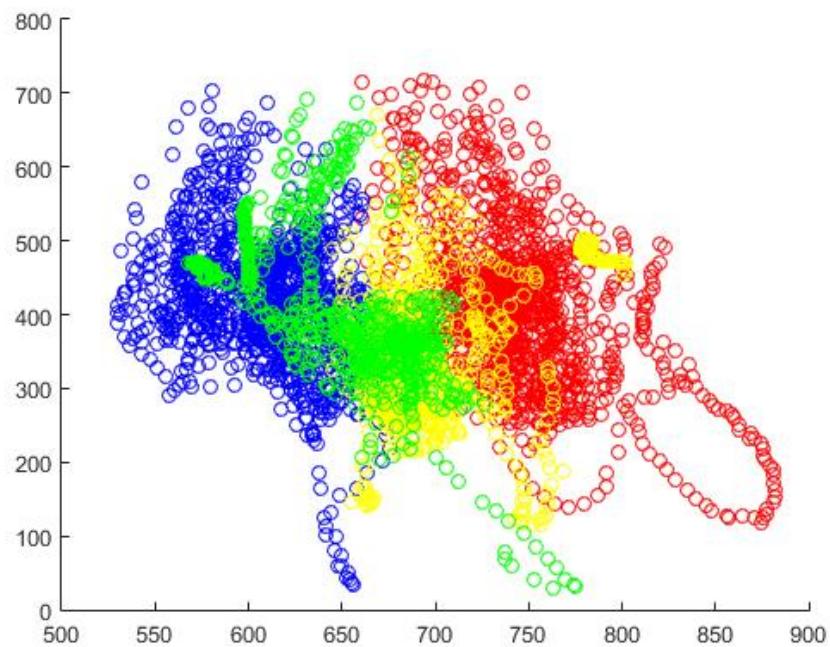


Figure 8.6: Front Left Camera Scatter Plot

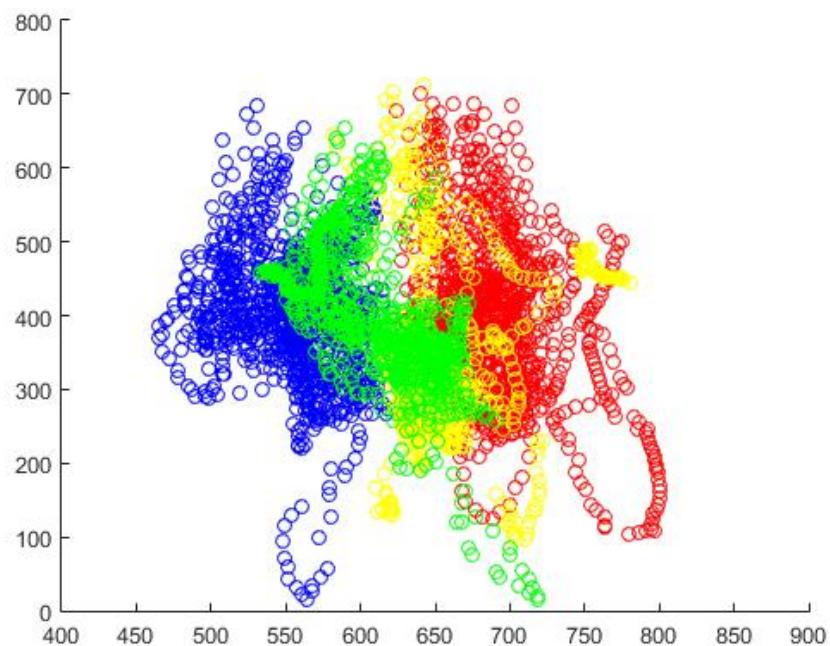


Figure 8.7: Front Right Camera Scatter Plot

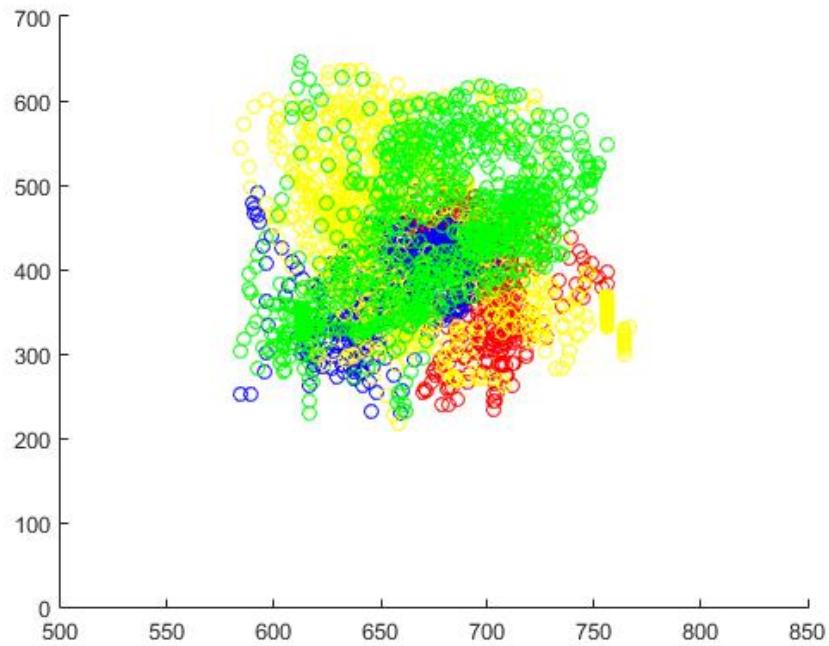


Figure 8.8: Back Left Camera Scatter Plot

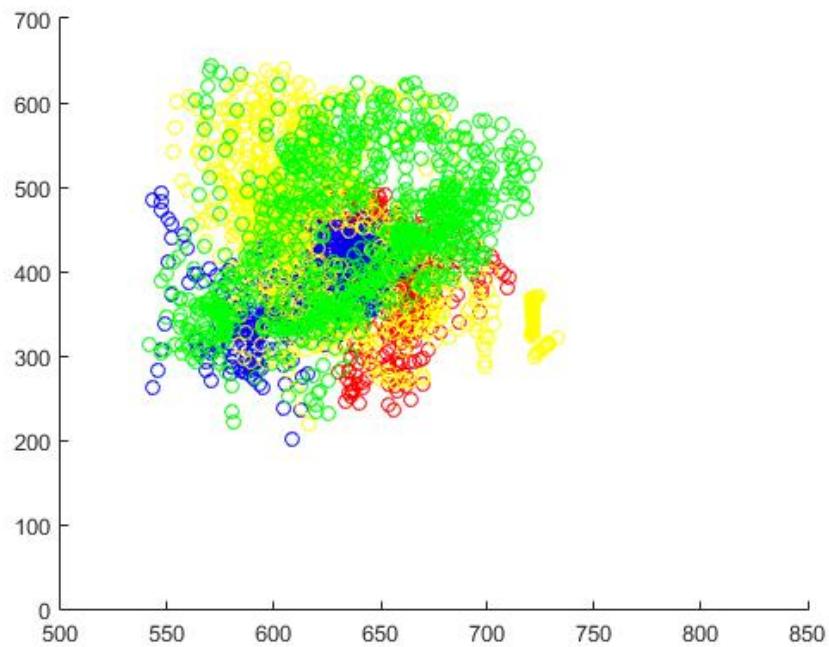


Figure 8.9: Back Right Camera Scatter Plot

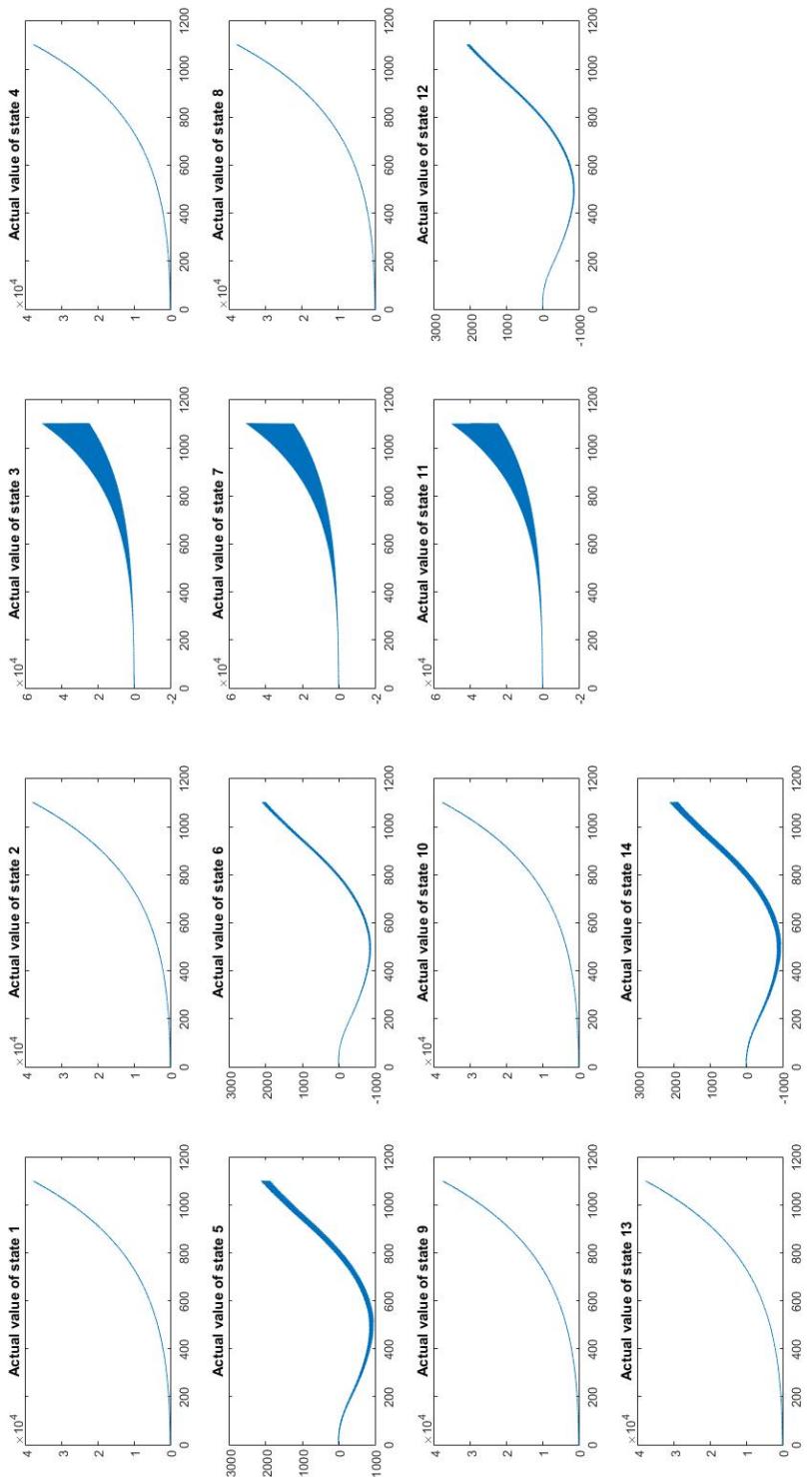


Figure 8.10: EKF Configuration 1 - Actual State Values

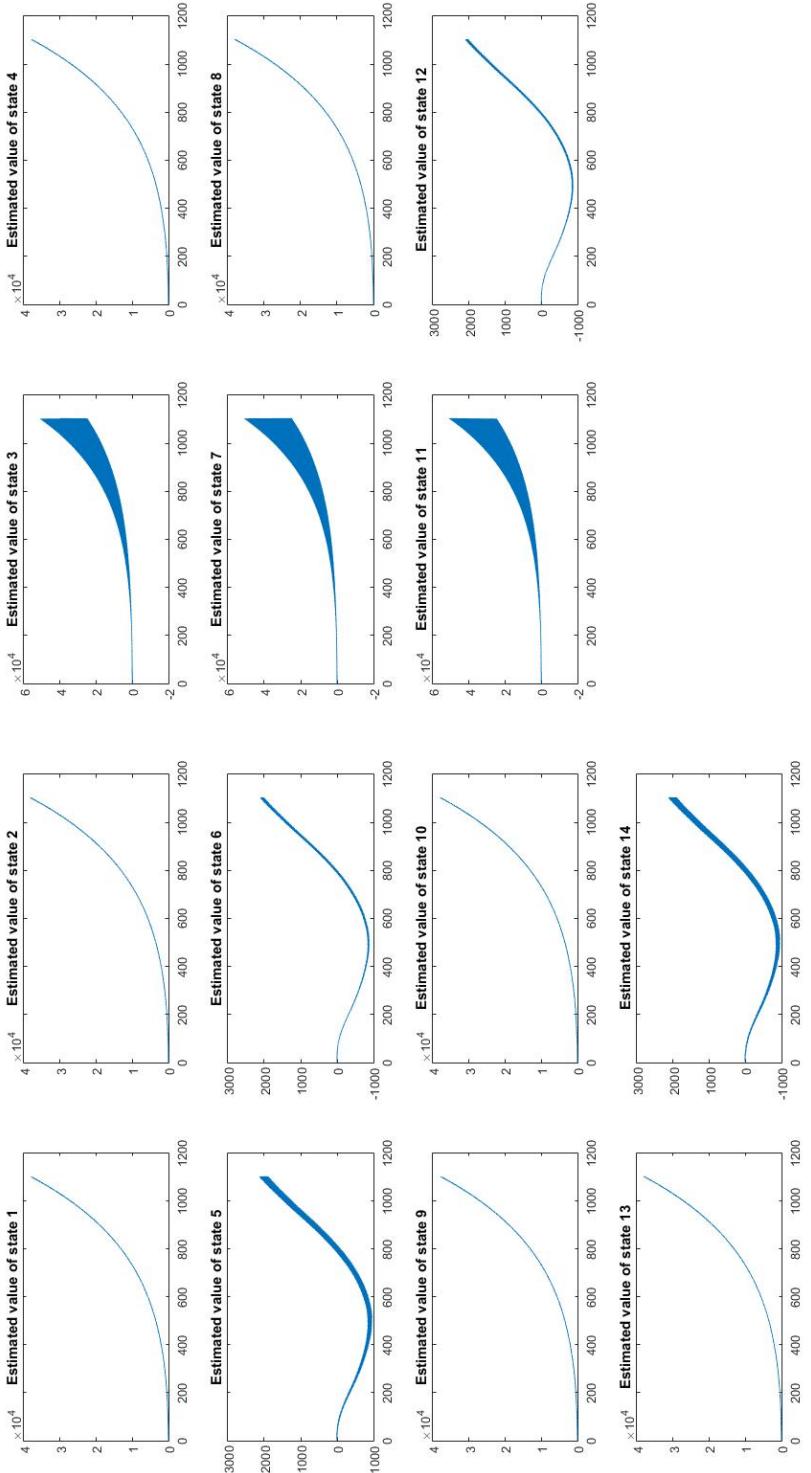


Figure 8.11: EKF Configuration 1 - Estimated State Values

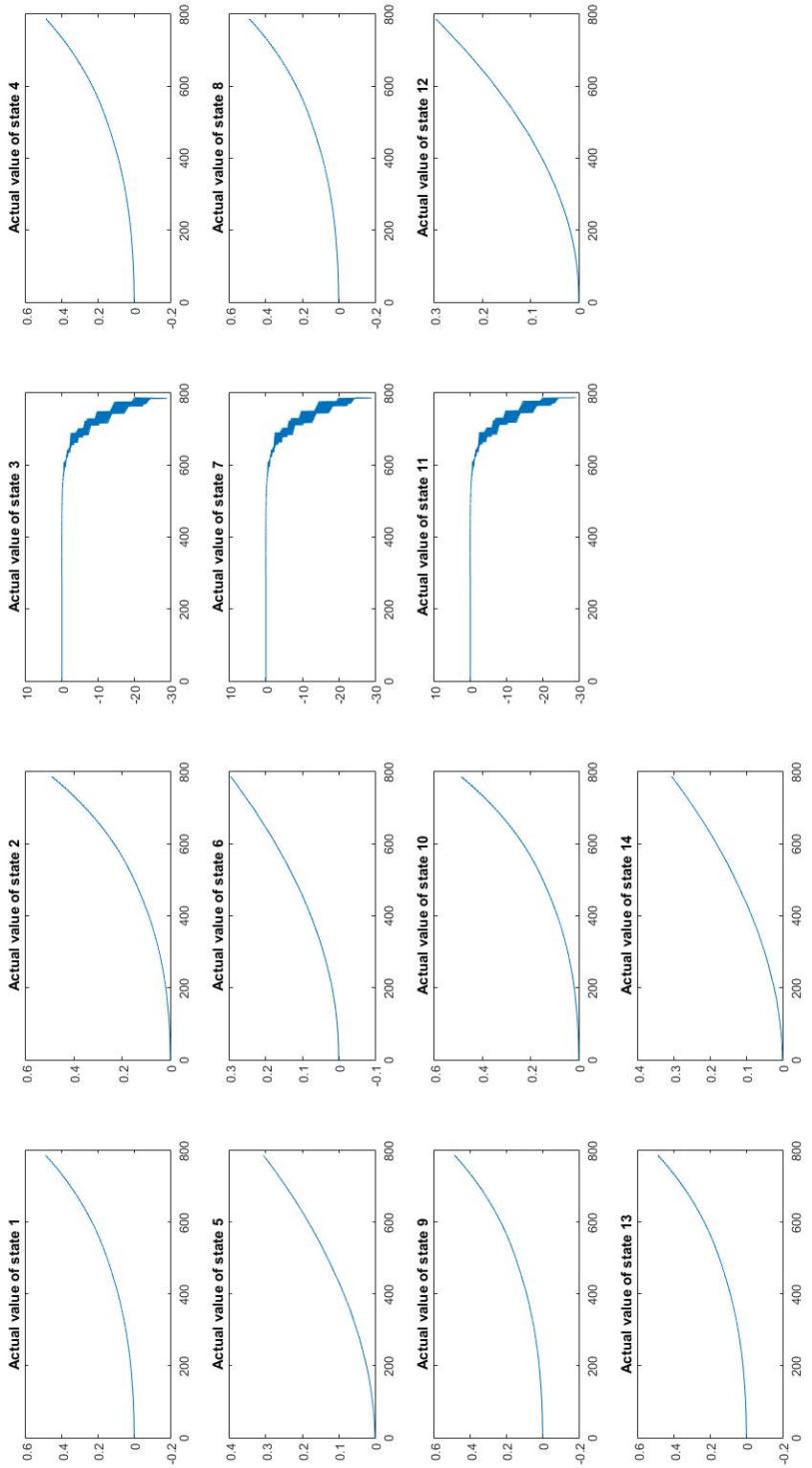


Figure 8.12: EKF Configuration 2 - Actual State Values

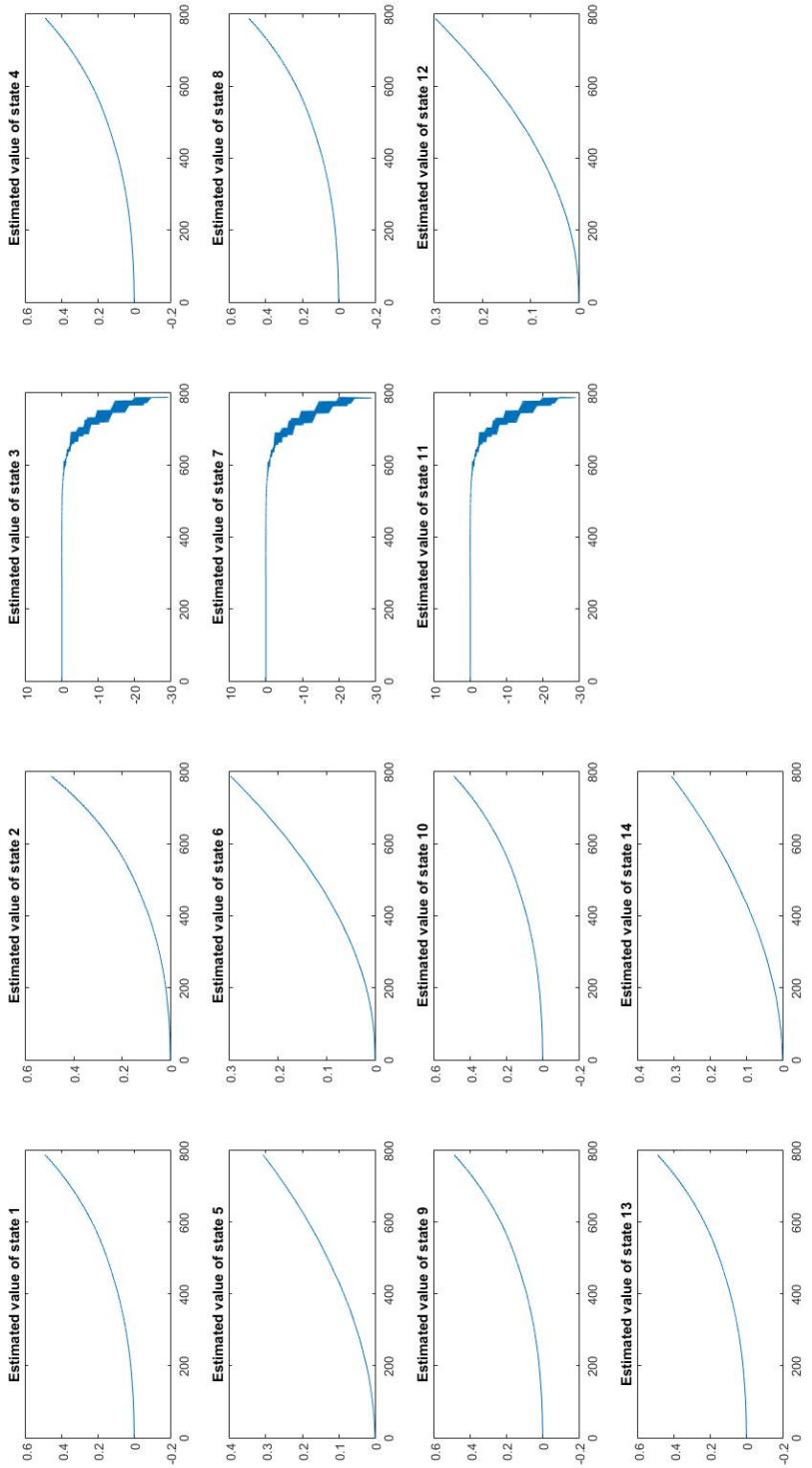


Figure 8.13: EKF Configuration 2 - Estimated State Values

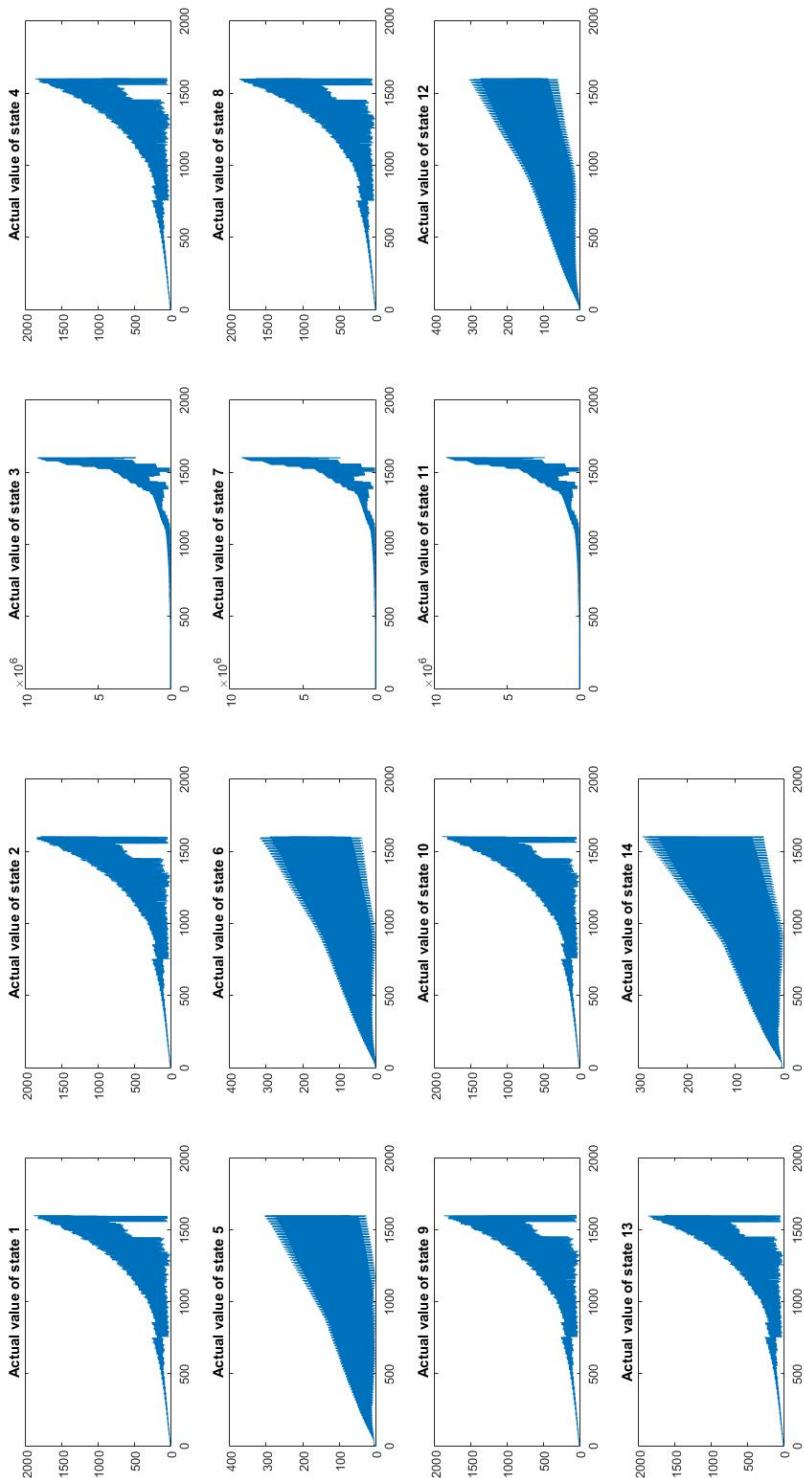


Figure 8.14: EKF Configuration 3 - Actual State Values

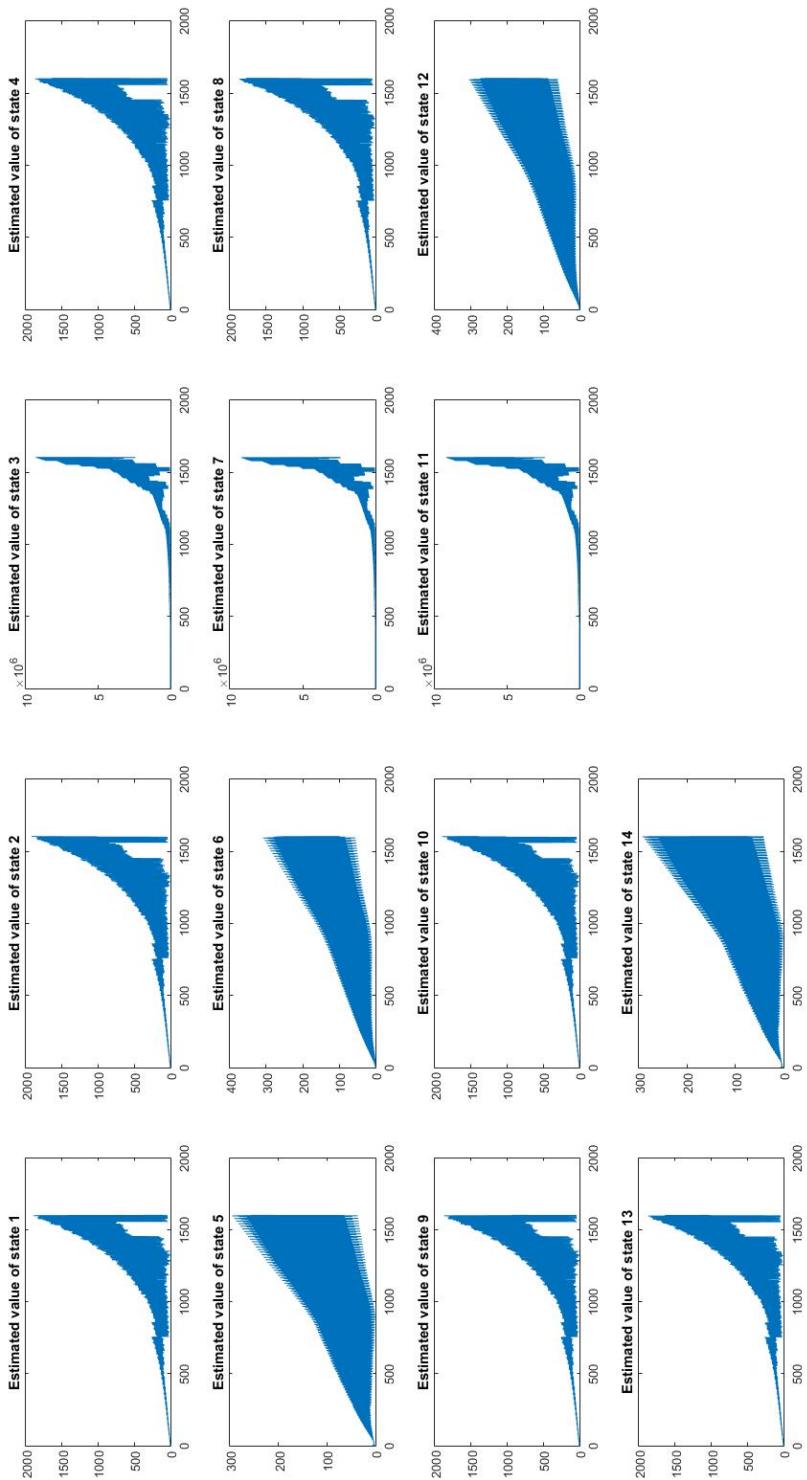


Figure 8.15: EKF Configuration 3 - Estimated State Values

## Appendix C: CD Guide

---

The accompanying CD is mentioned throughout the document. The CD has the following file structure:

CD/:

code\_android - Containing the smartphone application for data logging.

code\_latex - Contains the various TEX files to build this thesis PDF.

code\_matlab - Contains all MATLAB files created for this thesis.

data\_sensor - Contains the sensor data from various tests.

data\_videos - Contains the video data from various tests.

design\_solidworks - Contains all SOLIDWORKS files used in this project.

## Appendix D: Ethics and ELO Documents

Application for Approval of Ethics in Research (EiR) Projects  
Faculty of Engineering and the Built Environment, University of Cape Town

### APPLICATION FORM

**Please Note:**

Any person planning to undertake research in the Faculty of Engineering and the Built Environment (EBE) at the University of Cape Town is required to complete this form **before** collecting or analysing data. The objective of submitting this application *prior* to embarking on research is to ensure that the highest ethical standards in research, conducted under the auspices of the EBE Faculty, are met. Please ensure that you have read, and understood the **EBE Ethics in Research Handbook**(available from the UCT EBE, Research Ethics website) prior to completing this application form: <http://www.ebe.uct.ac.za/usr/ebe/research/ethics.pdf>

APPLICANT'S DETAILS		
Name of principal researcher, student or external applicant		Johann Hendrik Joosten
Department		Electrical and Electronics Engineering
Preferred email address of applicant:		joostenhendrik@gmail.com
If a Student	Your Degree: e.g., MSc, PhD, etc.,	B.Sc(Eng) Mechatronics
	Name of Supervisor (if supervised):	Dr. Amir Patel
If this is a researchcontract, indicate the source of funding/sponsorship		-
Project Title		Modelling the Kinematics of the Human Lower-Limbs using Cameras and an IMU

**I hereby undertake to carry out my research in such a way that:**

- there is no apparent legal objection to the nature or the method of research; and
- the research will not compromise staff or students or the other responsibilities of the University;
- the stated objective will be achieved, and the findings will have a high degree of validity;
- limitations and alternative interpretations will be considered;
- the findings could be subject to peer review and publicly available; and
- I will comply with the conventions of copyright and avoid any practice that would constitute plagiarism.

SIGNED BY	Full name	Signature	Date
Principal Researcher/ Student/External applicant	Johann Hendrik Joosten		22/08

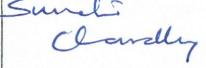
APPLICATION APPROVED BY	Full name	Signature	Date
Supervisor (where applicable)	Amir Patel		22/08/2017
HOD (or delegated nominee) <small>Final authority for all applicants who have answered NO to all questions in Section1; and for all Undergraduateresearch (Including Honours).</small>	S. CHOWDHURY		11/10/17

Figure 8.16: Ethics clearance

EEE4022F/S Final Year Project		Student Name: JOHANN HENDRIK JOOSTEN	Student Number: JSTJ0H004	State whether DPR/DPR is awarded: <input checked="" type="checkbox"/> DPR <input type="checkbox"/> DP
Supervision ELO Tracking Form 2017		Supervisor Comment	Supervisor Comment	Supervisor Comment
ELO 3: Engineering Design	Perform creative, procedural and non-procedural design and synthesis of components, systems, and/or processes.	Complete		
ACTION REQUIRED FROM THE STUDENT	Design and construction of a wearable data capture harnesses housing 4 cameras, oriented correctly, and an IMU mounted rigidly to a set of data capture cameras. Design of Estimation algorithm and various software programs to manipulate and interpret captured data.			
ELO 4: Investigations, experiments and data analysis	Demonstrate competence to plan and conduct investigations and experiments.	Complete		
The balance of investigation and experiment should be appropriate to the discipline. Research methodology to be applied in research or investigation where the student engages with selected knowledge in the research literature of the discipline.				
Note:	An investigation differs from a design in that the objective is to produce knowledge and understanding of a phenomenon and a recommended course of action rather than specifying how an artifact could be produced.			
ACTION REQUIRED FROM THE STUDENT	Planned approach to design and research within chapter 2 of the report titled "Methodology". Compiles a full review of literature pertaining to methods used in the research within chapter 2 of the report titled "Literature Review".			
ELO 5: Professional and technical communication	Demonstrates competence to communicate effectively, both orally and in writing, with engineering audiences and the community at large. This course evaluates the long report component of this outcome at exit level.	Complete		
Material to be communicated is in an academic or simulated professional context. Audiences range from engineering peers, management and lay persons, using appropriate academic or professional discourse. Writing requirements range from short (100-1000 words plus tables/diagrams) to long (10 000 to 15 000 words plus tables, diagrams and appendices), covering material at exit-level. Methods of providing information include the conventional methods of the discipline, for example engineering drawings, as well as subject-specific methods.				
ACTION REQUIRED FROM THE STUDENT	Write report to serve as proof of writing proficiency. Perform oral exam to serve as proof of verbal communication proficiency.			
ELO 6: Individual, team and multidisciplinary working	Demonstrates competence to work effectively as an individual, in teams and in multidisciplinary environments. This course evaluates the individual working component of this learning outcome at exit level.	Complete		
ACTION REQUIRED FROM THE STUDENT	Perform research, design, experimentation and data analysis individually with limited input from project supervisor. Draw conclusions based on final results. Verify result from existing literature and verification tests.			
ELO 9: Independent learning ability	Demonstrate competence to engage in independent learning through well developed learning skills.	complete		
Operate independently in complex, ill-defined contexts requiring personal responsibility and initiative, accurately self-evaluate and take responsibility for learning requirements, be aware of social and ethical implications of applying knowledge in particular contexts.				
ACTION REQUIRED FROM THE STUDENT	Differentiate between critical and less important elements of research, apply self taught concepts and follow ethical research procedures. Defining a scope for a complex problem space.			
Student Name	Johann Hendrik Joosten	Internal Examiner Name		
Student Signatures	Final Year Student of Engineering	Internal Examiner Signatures		
Designation	Dates	Designation		
	24-Oct-17	Dates		
		Internal Examiner Dates		
		Dr Amir Patel		
		Internal Examiner		
		Designation		
		Dates		
		24-Oct-17		

Figure 8.17: ELO clearance