Gait Phase Detection of Stair Ambulation using Inertial Measurement of Lower Limb

Michael Stanley
B.Eng. (Hons), Monash University, Australia

A thesis submitted for the degree of *Doctor of Philosophy* at
Department of mechanical and Aerospace Engineering
Monash University
in 2021

Copyright notice

Notice 1

*The second notice certifies the appropriate use of any third-party material in the thesis. Students choosing to deposit their thesis into the restricted access section of the repository are **not** required to complete Notice 2.*

Notice 2

I certify that I have made all reasonable efforts to secure copyright permissions for third-party content included in this thesis and have not knowingly added copyright content to my work without the owner's permission.

New page

Abstract

Existing wearable assistive devices were primarily designed for assisting motion that is physically challenging for the user. The lower-limb devices have been mainly designed for the most common activity of walking. Comparing to level walking gait, a smaller group of studies have considered the detection and control scheme for progressing stair gaits, which is also vital for a user to maintain an independent lifestyle.

Gait phases are described in gait analysis, and gait phases detection techniques are derived from the standard gait analysis, which primarily captures the user's kinematics and kinetics. IMUs are the most viable technology to incorporate into wearable devices to capture the user's gait kinematics. It is widely used in literature due to its commercial viability, physical robustness, and user-friendliness in deployment. An ambulatory sensory system is developed from a commercial knee brace integrated with IMUs and using footswitches as ground truth. The angle measurement of the IMUs is verified to be statistically consistent with an encoder-based system in tracking the same motion.

The thesis presents a real-time adaptive parametric rule-based gait phase detection approach for stair ambulation using kinematics measurement in the core study. The study addresses the lack of functional gait phase detection technique suitable for real-time application for stair gaits. A successful detection method would lead to future research that could develop an effective control scheme for assistive devices to provide timely assistance to the user during stair gaits. The method is validated by an experiment with 20 participants wearing the modified ambulatory system. The performance is analysed using F1-score for reliably detecting the gait phases, using statistics of the timing error for its timeliness in detecting each gait phase, and the usefulness of the method by evaluating the likelihood of an unacceptable time error. The experiment tests the detection in its intended operational environment over a staircase with multiple progressive steps. The results support the reliability and usefulness of the implemented approach. It results a high overall F1-score of 0.9925 with an average error below 50 ms.

The experiment provides valuable data on stair gaits in an out-of-lab environment, which open up other post-hoc studies. This thesis includes a comparison study with detection based on common machine learning techniques using existing data.

The rule-based algorithm achieved a high overall F1-score of 0.9925 across the 7 selected phases of both stair ascent and stair descent. The worst F1-score of 0.9670 is occurred at the Controlled Lowering phase during stair descent, whereas the best is F1-score of 1.0 is achieved by the Foot Placement phase during stair ascent. The detection has a mean timing error [standard deviation] of 43.25[30.21], 20.12[15.23], -30.17[23.43], and -43.66[16.41] ms for ascent IC, descent IC, ascent EC, and descent EC respectively, where negative errors are representing delayed detection.

For the 36 neural network models trained, 3 different optional filtering is applied to the output to stabilize the output classification. Also, there are 118 and 119 supervised classifiers trained for stair ascent and descent respectively. None of trained networks or classifiers outperform the ruled-based algorithm in all aspect of detection performance for all phases. Outperforming machine learning models are present for a specific phase in either the F1-score or in timing error or the consistency of the timing error. The study found it is possible to deploy the trained models to complement the performance of other models or the rule-based approach presented for the detection of a specific phase. The performance trends related to the type of training parameters are recorded in the thesis, and the results could provide a guide for other researcher and developers to follow in choosing the appropriate model for their application, making the appropriate trade-off in performance, and choose the models to complement each aspect of performance.

Declaration

This thesis is an original work of my research and contains no material which has been accepted for the award of any other degree or diploma at any university or equivalent institution and that, to the best of my knowledge and belief, this thesis contains no material previously published or written by another person, except where due reference is made in the text of the thesis.

Signature: ..Michael.....

Print Name: ....MICHAEL STANLEY

Date: ..16/04/2021..

New page

Acknowledgements

Completing a PhD is a unique experience, I learnt so much and yet much more to learn. It is a voyage of curiosity, discovery, and perseverance. Researchers have spent astronomical amount of effort to explore each corner of the unknown knowledge. They are challengers, pioneers, and truth seekers, finding consistency in a world of uncertainty and sharing discovered knowledge to a wider community. The academics community has all my respect and admiration, and I felt honoured to have the opportunity to present myself as a part of this selfless and fearless community. I am thankful and would like to express my gratitude to each and everyone who had supported me and the thing that I hold dear here.

The scientific method used in conducting and reporting research is a refined and systematic process that allow outcomes to be reproducible and validated with statistically significance. It is an elegant process that is the results of the endless pursue of truth by many great minds over centuries. I am grateful that this system is a known knowledge to me, and thanks the philosophers who had made this possible by challenging the logics back at their times. Each philosopher adds their own thinking process into the method itself. We all built upon existing knowledge and challenge the existing concept with following up studies that try to fill the gaps and limitation of establish knowledge. All works are collaboration with people who came before us. In that, I am grateful to the entire academic community.

The acknowledgment section is one section where researchers could express their sincere thought freely. I viewed this section as an oasis among the wall of text that follows it, where the author could be themselves as a person without the need to be uphold the objectivity required for reporting scientific outcomes. In that, I appreciate the tradition of writing acknowledgment. My first specific thanks would go to this acknowledgment section. I thank you for allowing my true self to be presented, and I thank you for allowing me to leave a remark without any restriction.

In my specific journey, I would like to thank all my supervisors for their guidance, advices and assistance. I appreciate the days we spent on discussing and exploring the different opportunities. The achievement of this thesis is product of your patience and your kindness. Dr Chao Chen, Dr Anna Murphy, Prof Flavia Cicuttini, and Prof Marcus Pandy, you all have my deepest heartfelt "Thank you". I could not appreciate your support more, words itself has limited my ability to express it fully.

Never have a meeting been unfruitful, my mind is stimulated and my thoughts are refined. Each step going forward, I found myself at a new height with your helps. It strengthens my research foundation to give it a clearer purpose, not just to the specific area of research but also the greater picture. It is you and your efforts that make this research possible. You are my role models and mentors, I look up to each one of you, hoping that one day I will reach your height on my own hereafter. Thesis never is the end of between the PhD candidates and their supervisors, it is a new beginning. We are going to move forward as a greater community. Thesis is an opportunity to give back, and perhaps let others to expand upon. I hope this work would inspire others to follow the path in the same way you had inspired me to do so. It is my turns to be the giver and be patient. The work is going to pass down my knowledge to the seeker, just the same way you have passed down yours to me.

Apart from the academics support I received mainly from my supervisor and historical thought leaders. I declare that "This research was supported by an Australian Government Research Training Program (RTP) Scholarship". The financial support allows me to focus on producing scientific results with high standard and integrity without the distraction of taking additional income sources to maintain basic human needs. I also understand that the financial support has also funded the PhD program with adequate facility and service within the university so that my work is conducted properly.

At last, my thanks would go to my friends and family who has supported me through the PhD program. A special thanks go to my cat "Lucky", who often visited my room and provide the warmth and comfort I need over the cold winter nights and mornings. I would thank my colleague and friends in my research group, where we often bounce idea off each other, and verify the logics in our idea. Special mention goes to Dr Godfrey Keung, Dr Shao Liu, Dr BinBin Chen and others who share many of my hobbies. Having friends who I can talk to at a personal level has been helpful in my mentality. Having friends who had walked the path and can sympathise with me has allowed me to be resilient throughout my journey.

Table of Contents

CHAPTER 1 : Introduction 12
Key Contributions: 15
Thesis Organisation: 16
CHAPTER 2 : Literature Review 18
Overview: 18
2.1. Gait Phase Detection 18
2.2. Biomechanics of Stair Ambulation 23
2.3. Gait Analysis 26
2.4 Research Problems and Aims 29
CHAPTER 3 : System Description 30
Overview 30
3.1 Wearable Ambulatory System 30
3.2 IMU Background 32
3.3 IMU Performance 45
CHAPTER 4 : Adaptive Real-time Detection Algorithm 59
Overview 59
4.1 Algorithm Development 59
4.2 Experiment 68
4.3 Data Analysis 69
4.4 Results 70
4.5 Discussion 73
4.6 Conclusion 77
CHAPTER 5 : Machine Learning Approaches 79
Overview 79
5.1 Data Preparation 79
5.2 Data Analysis 80
5.3 Time Series Neural Network 81
5.4 Supervised Learning 97
5.5 Conclusion 122
CHAPTER 6 : Conclusion and Future Works 123
Reference 128
Appendix 137
A: Eulers Angles Conventions 137
B: Quaternion Rotation: Matrix Formulation 138
C: Accuracy of Supervised Learning Models 140

Table of Figures

Figure 2.1: The average hip flexion angle during one gait cycle of stair ascent (black) and descent (blue) across 8 subjects. Vertical lines are the transition between the gait phases. Phase codes are defined in the text. Diagram is redraw from data provided in [104] 25

Figure 2.2: The average knee flexion angle of one gait cycle of stair ascent (black) and descent (blue) across 8 subjects. Vertical lines are the transition between the gait phases. Phase codes are defined in the text. Diagram is redraw from data provided in [104] 26

Figure 2.3: The average ankle dorsiflexion angle of one gait cycle of stair ascent (black) and descent (blue) across 8 subjects. Vertical lines are the transition between the gait phases. Phase codes are defined in the text. Diagram is redraw from data provided in [104] 26

Figure 3.1: Measurement system: (a) the knee brace with IMU attached to the centre of each green circle, red: Y axes, blue: Z axes; (b) insole footswitches, green square indicates the location of each contact area. 31

Figure 3.2: Top view of the IMU: X-axis (Roll) points to the right of the page, Y-axis (Pitch) points to the top of the page, Z-axis (Yaw) points out of the page. 33

Figure 3.3: A 3D representation of an IMU rotation, Green arrow is the gravity reaction force measured by the IMU, and oranges are the gravity reaction force projection in the IMU transformed frame which are the acceleration measured by the IMU in each axis. XYZ is the world frame, xyz is the IMU original frame, x'y'z' is the IMU transformed frame. 40

Figure 3.4: Flow chart diagram of a Kalman Filter operation 44

Figure 3.5: Yaw angle of the calibrated Shank IMU including the first 5 minutes after startup 47

Figure 3.6: Experiment setup of the IMU performance tests. Orange: the location of the actuation which drive the shank segment up and down (the red arrows). Purple: the location of the encoder to measure the angle difference between the brace's segment. 48

Figure 3.7: Static noise of Yaw-Pitch-Roll angles of the calibrated (top) and the controlled (bottom) IMU 50

Figure 3.8: Standard deviation of the static trials of the calibrated IMU. 51

Figure 3.9: The mean and one standard deviation of the maxima of the IMU and encoder measurement across the ten five-minute intervals over the five hours 55

Figure 3.10: The mean and one standard deviation of the minima of the IMU and encoder measurement across the ten five-minute intervals over the five hours 56

Figure 4.1: Figure shows the angle ψ captured by IMU in the thigh and shank section of the brace. The red arrows indicate the Y axes, the blue arrows indicate the Z axes. 60

Figure 4.2: Top-level flowchart diagram of the algorithms: from the sampling and derivatives of input data to the calibration stage, and real-time detection and step-wise update. The input data are the current timestamp, $t(n)$, and measurement, $\psi i(n)$, where n is the program cycle, and i is type of variable 61

Figure 4.3: The calibration stage: Maximum and minimum of the ψt, ψs, and $\psi \Delta$ and their derivatives are continuously monitored until one complete step is detected after all other turning points are found in other variables. 62

Figure 4.4: A real-time detection example: the algorithm uses previously established parameters to search for the first instance of event occurrence within the threshold window. 63

Figure 4.5: The real-time detection and update stage: the sampling data are checked with both the event feature and the threshold windows. A missed event is flagged if no event is detected within the entirety of the threshold. A successful detection in both will update all parameters and thresholds accordingly. Gait cycle time update when next $\psi\Delta$ maximum is detected. 65

Figure 4.6: A participant walking down a staircase in an out-of-lab environment wearing the measurement brace 68

Figure 4.7: The detection performance of the algorithm for each selected phase. TPR = true positive rate (recall), TNR = True negative rate, PPV = Positive prediction value, NPV = negative prediction value 71

Figure 4.8: The average timing performance of the algorithm for each reference event (IC or FO) in stair ascent across all subjects. Error bar of each bar represents the standard deviation. 72

Figure 4.9: The average timing performance of the algorithm for each reference event (IC or EC) in stair descent across all subjects. Error bar of each bar represents the standard deviation. 72

Figure 4.10: An example of IC instance of subject 3, where the IC instance is close to the minimum of $\psi\Delta$. 75

Figure 4.11: An example of IC instance of subject 6, where the IC stance is close to the minimum of ψs 75

Figure 5.1: One gait cycle of the different output types with thigh roll, shank roll, and knee flexion angle. 80

Figure 5.2: The network structure of a 20 delay and 10 neurons NARX network. This diagram is a generation with view(net) command on Matlab using Simulink diagram block. 81

Figure 5.3: The network structure of a 2-delay and 10 neurons NIO network. This diagram is a generation with view(net) command on Matlab using the Simulink diagram block. 82

Figure 5.4: Scatter plot of the original data for the two 5-sample wide transition outputs (initial contact and foot off) during stair ascent with axes presenting the angle and velocity of each attached body (top: thigh, middle: shank), and the presumed joint (bottom) 120

Figure 5.5: Scatter plot of the original data for the two 5-sample wide transition outputs (initial contact and foot off) during stair descent with axes presenting the angle and velocity of each attached body (top: thigh, middle: shank), and the presumed joint (bottom) 121

Table of Tables

Table 2.1: Summary of GPD performance in the literature using inertial measurement 22
Table 2.2: Gait partitioning of stair ambulation 23
Table 3.1: Options for configuration of data rate and digital low pass filter of MPU6050 ... 32
Table 3.2: Mathematics equations of Kalman filter in each operation 43
Table 3.3: Summary of the statistical measure of the static trials 49
Table 3.4: Intertrial statistical results 53
Table 3.5: Results of continuous test between the IMU and the reference encoder system. 54
Table 3.6: Normality test of randomly distributed random values of 10,000 samples with discretised and truncated condition. 57
Table 4.1: Kinematic events for the detection of gait phases 60
Table 4.2: Conditions of critical points 66
Table 4.3: Conditions of updating turning points in calibration 67
Table 4.4: Participants Profiles 68
Table 4.5: Detection Timing Errors and Variations 73
Table 5.1: A summary of all configuration of trained NN models 83
Table 5.2: Time series NN performance for initial contact of stair ascent 85
Table 5.3: Time series NN performance for end contact of stair ascent 87
Table 5.4: Time series NN performance for initial contact of stair descent 89
Table 5.5: Time series NN performance for end contact of stair descent 92
Table 5.6: Average timing performance of the model with F1-score above 0.9 95
Table 5.7: A summary of all configuration of trained models 98
Table 5.8: Supervised learning classifier performance for initial contact of stair ascent .. 101
Table 5.9: Supervised learning classifier performance for end contact of stair ascent 105
Table 5.10: Supervised learning classifier performance for initial contact of stair descent 109
Table 5.11: Supervised learning classifier performance for end contact of stair descent 113

CHAPTER 1 : INTRODUCTION

Exoskeletons and wearable robots have found many applications in different industries, including human augmentation [1, 2], bodyweight support [3, 4], remobilising of limbs [5]. They are multidisciplinary outcomes of combing many technologies. Recent advancements are enabled by the minimisation of the sensors, the actuators and the computational processing units. It also allows a complex system with multi-axial control to be portable and responsive.

Agostino *et al*. [6] had outlined a list of area of possible technological advancement to have a safe and dependable physical human-robot interaction (pHRI) between the exoskeleton and the user. Many of the progress made since that are aligned with their vision. Some devices [1, 7-9] are incorporating serial elastic actuators (SEA) for back-drivability, force controllability using the elastic component and better mimic the natural human joints. The actuation of joints is often driven by the robots [10, 11] but initiated by the user. The user often has to select the modes and trigger the action using an interface [12]. Some system uses admittance control to trigger predefined action [13, 14]. Other controls strategies are discussed in details in [15] and [16].

One major application of exoskeletons is weight carrying, such as Honda weight support, HAL [17] and BLEEX [18]. They designed to transmit the additional load through the external structure bypassing the human user. They allow the user to lift more weight for a longer duration than usual. The control scheme varies between the devices. Honda weight support controls the force output to offset a portion of your body weight. HAL allows the user to control the direction of motion through sensing surface bioelectrical signals. BLEEX motions are driven by the robot while relying on the user for balancing. Most active exoskeletons do not consider the user's intention as they do not react to the user input kinematics direction. The development of HAL5 [19], which has included assistance for patients with a disability such as paraplegia, incorporates the sensing of body kinematics and posture control. Nevertheless, the motion is still driven by the robot instead of the human.

Medical application is another major active research area of exoskeletons and wearable assistive devices. The medical robots market is projected to reach USD 12.7 billion by 2025 from an estimated USD 5.9 billion in 2020 at a compound annual growth rate (CAGR) of 16.5% [20]. Exoskeleton in medical applications could serve different purposes, including

addressing symptoms of diseases such as osteoarthritis [21] and muscle weakness, rehabilitation training [10, 22], clinical measurement instrument [23]. An active exoskeleton is not commonly used here since these applications focus on studying or improving the existing human locomotion. Replacing the human's gait with that of the robot would counteract many existing treatments to gait disorder. It is essential for these devices to cooperate safely with humans so that the robot does not apply additional stress to the user, nor does it under-provide for the required motion. Hence, the system has to react appropriately based on the current movement.

Among existing commercial systems, C-Brace® [24], ReWalk™[25] and HAL® [26] have been developed as a medical application for lower-limb impairment. Aside from remobilising paralysed patients, these devices have limited application. They are not suited to users with a less extreme disability, where the motion is preferred to be driven by the user. The device provides assistance as needed to either support the user or to correct the gait. That is an application with room for development and significant impact. For example, osteoarthritis (OA) alone would be a multi-billion-dollar industry worldwide. Australia alone expects 3 million people affected by the disease with knee osteoarthritis being the most common form [27] by 2032 from a report dated back in 2013 [28]. With limited mechanical intervention that slows down the disease's progression, such as cane and walker, a robotics device that could assist patients effectively would be a game-changer and a high impact outcome. It also potentially releases valuable availability of surgery room for other diseases by postponing total knee replacement surgery and millions of savings in the economy by avoiding the second surgery completely. Another example of wearable lower limb devices in medical environments includes usage as a measurement system [29], and as gait training devices [30-32]. Lower limb assistive devices would be the focus of the current study since it has a clear application to be addressed.

The biggest hurdle for patients to accept wearable mechanical interventions is the lack of control and uncomfortableness of existing devices, as found by a survey [33]. It is expected that a system that could recognise the user's motion and then apply timely assistance to reinforce it can alleviate the misalignment in motion between the machine and the user. An ideal assistive device would recognise the user's intention and allow the user to move toward the intended direction with support against external forces.

From the control prescriptive, the ability to detect gait phases is crucial to the control processes to reinforce the user's movement [34] and enable the correct classification of gait phases within an acceptable range of error [35]. A mistake of ±50 ms is deemed acceptable for many biomechanics applications [36, 37]. There is a need to develop assistive devices that could accurately detect the user's gait and react accordingly. This study is intended to address this need by investigating the detection and classification of gait phases that facilitate this human-machine interaction.

In the field of biomechanics, the gait of each activity is generalised and classified into different phases depending on the function of the limb during each phase of the gait. A system that could accurately detect the transition between the phases would provide the required assistance to the user based on each respective phase. The gait phases are defined specifically for each activity, such as level walking, stair ambulation, sit-stand transition. Hence, gait phase detection (GPD) is typically tailored to each type of activity.

There are many well-documented approaches to gait phase detection in literature; refer to chapter 2.2. The majority of them are offline post-analysis and gait parameterisation, and the algorithms' real-time performance remains largely unverified. Since their aim is to enhance the gait analysis process, the accuracy and reliability of the detection is their primary objective. These solutions are often impractical to be implemented on wearable devices under real-time control with extensive use of high precision sensors such as force plate, EMG, and vision-based motion capturing systems. This study is aimed to transfer the knowledge on gait biomechanics to detect critical phases on a wearable device, to bridge the gap between biomechanics and robotics engineering.

Walking is the most common form of activities of daily life (ADL). It has been studied extensively in biomechanics, and most lower-limb assistive devices are designed to assist this activity. Some existing studies have provided a working approach in real-time gait phase detection on wearable devices. In comparison, fewer studies have examined stair ambulation, yet the capacity to undertake this skill is essential in maintaining independent function. The lack of detailed study of stair ambulation GPD is limited by the availability and capacity of an instrumental staircase. Therefore, there is a lack of proven working examples of a real-time gait-phase-detection for stair in an out-of-lab environment.

Furthermore, the stair ambulation is joint closely tied to the patellofemoral joint. This join is also a medical treatment gap against knee OA [38, 39]. Patients with patellofemoral OA typically experience pain during stair ambulation [40, 41]. Surgery of total knee replacement is targeting the tibiofemoral joint, not the patellofemoral joint [40]. It is clinically recommended to avoid the activity altogether while conducting knee joint exercise therapy [42]. Total knee replacement with patellar resurfacing is the only predictable positive outcome; however, it is an aggressive approach and not recommended for single compartment disease [43]. An effective alternative treatment is yet to be discovered and much needed.

Key Contributions:

This work contributes toward three major areas: 1) the technological advancement of GPD on wearable assistive devices, 2) the expansion of the biomechanical data of stair ambulation on a real staircase, and 3) the deployment of real-time GPD in its potential applications.

1. Technology advancement of GPD on wearable assistive devices

An outcome of this work is the validation of a real-time GPD on an actual wearable device. It contributes to the technological advancement of wearable devices by providing a functionally verified example. It would also attempt to develop a performance evaluation that would provide quantitative results as a benchmark for future researchers and developers. It is crucial to have an accurate and reliable detection algorithm to enable appropriate assistance for gait progression without significant discomfort to the user. A proven system that could capture the kinematics of the user reliably could lead to the development of a wearable measurement instrument at the same level of accuracy as an optical motion capturing system, the current golden standard in a gait analysis laboratory.

2. Deployment in real-time on a wearable structure

This study would capture a valuable set of real-time data of stair gait on an out-of-lab staircases. Verifying the consistency of the motion capturing would prove useful for IMU to be used as a reliable method of measuring gait kinematics on a wearable structure. Data collected from the experiment would build a database with thousands of steps, which could be used by others in the community to develop their approach. Biomechanists could use the

information regarding the implementation to reproduce a wearable measurement system and verify it with standard laboratory setup. A verified wearable measurement system would allow out-of-lab experiment to be reliable as the laboratory setup, thus it provide a reliable option for biomechanism study in different terrain and conditions to be conducted with a wearable device instead of the setting up a complex system of sensor around in the environment or reproducing the complex environment inside a laboratory.

3. Deployment of real-time GPD in its potential application

The study is intended to test the developed detection algorithm in its potential application, where the wearable device helps the user to complete stair ambulation over a staircase with multiple progress steps. The experimental results would directly reflect the actual performance of the algorithm in its application since the testing environment is consistent with the intended application. The study would provide the technical details of the implementation process, which is not commonly recorded in the literature [12].

An implementation of a real-time stair GPD could be significant in medical treatment for patellofemoral OA. Patellofemoral pain generally occurs when walking up and down the stairs for patellofemoral OA patients. Therefore, a stair GPD could pave a way to develop an assistance scheme that alleviates the patients' pain. Note that clinical treatment with actual patients is outside the scope of developing an approach for real-time stair GPD algorithms.

Thesis Organisation:

Chapter 2 begins with investigating the existing and developing technology of GPD. It places a heavier focus on the sensor technology and the detection algorithms. It then examines the established biomechanics knowledge that acts as the foundation of how the gait phases are defined and partitioned in stair ambulation, the activity of interest in this study. This knowledge will enable the researcher to identify the ground truth and provide clear objectives that the GPD algorithms should achieve. Then, the chapter will investigate the existing measurement system and gait analysis methods used by biomechanists as the basis to develop the measurement system and the performance parameters to evaluate the GPD. Finally, it summarises the key research gaps and the problems this study is going to investigate.

Chapter 3 describes the system developed and used for the search project. It will provide the design, implementation and evaluation of the system. It provides detailed instructions for reproducing the system with expected benchmark performance.

Chapter 4 is the study of a rule-based algorithm developed during this research project. The detection rules are defined based on the description provided by studies on the normative stair gait. Then the system is tested on participants in a staircase with multiple flights of step, a realistic environmental that reflect the intended application

Chapter 5 utilises existing data to train models through machine learning to identify critical gait phases. Models include all available types in the "Classification learner" app and the NARX and NIO network in the "Neural Net Times Serie" app in Matlab2020a/b. The results of the machine learning approach will be compared with the developed rule-based approach in chapter 4. Future improvement is suggested from the limitation observed in the analysis.

Chapter 6 concludes the significant findings from the research project and summarise future research opportunities enabled by the findings of this research project.

CHAPTER 2 : LITERATURE REVIEW

Overview:

This chapter aims to provide the technical and theoretical information required to select, develop and evaluate the sensing system, the detection algorithm, and the analysis techniques. Section 2.1 presents the sensor technologies and detection techniques studies and used in literature. It is aimed to find out the most suitable technologies for wearable devices and the current gaps among the existing GPD approach. Section 2.2 describes the biomechanics of stair ambulation and the definition of each phase. The information would ensure that we are defining and detecting the phases consistent with the biomechanists. Section 2.3 describes the existing equipment and methods used in gait analysis. The information would be the basis of how our measurement devices, which described in chapter 3, is designed to capture and process the data consistent with the standard gait analysis. Section 2.4 summarises the research gaps and states the problem this study is going to investigate.

2.1. Gait Phase Detection

The first section explores the current state-of-the-art development in GPD algorithms. The first subsection reviews the existing sensor technologies used for existing GPD. The second subsection reviews the techniques used by the detection algorithm.

2.1.1. Technologies for Detection

Gaits phases are the partitioning and classification from gait analysis. The technologies used to capture the input required for the classification of phases are the same as gait analysis. Gait analysis facilities typically capture the kinematics of the body, the force acting on the bodies, and occasionally the bioelectrical signal from the EMG sensors.

Gait phase detection can be performed accurately with gait analysis systems which combine video motion cameras [36, 37, 44] and force platform [45, 46]. This is the golden standard for determining the gait phase. The force platform provides the ground reaction force data in 3 axes with a high frequency of reading, and it can identify the moment of initial contact

(IC) and foot off (FO) accurately. Modern motion capturing operates at and above 100 Hz, with displacement error of each marker under 1 mm within the configured space of the camera [47]. These two systems of equipment have been prominent in many biomechanics studies due to their proven and repeatable measurement [48]. However, these systems are not portable and cannot be used in conjunction with wearable devices.

Muscle electromyographic (EMG) activity has been used to detect the various phases during a gait cycle [49-51]. This technique could directly reflect how the body is behaving during motion. There are limitations to this approach as well. Firstly, EMG probes are sensitive to external force, and it is yet to be proven on a wearable assistive device. Secondly, the setup of EMG probes required specialist knowledge and calibration on each subject, which is difficult to deploy readily and quickly.

Footswitches and force-sensing resistors [52-54] offer an alternative design to detect IC and FO. These sensors can be attached under the feet to measure the presence of force. This type of sensor is used to provide the ground truth of IC and FO for many studies. Wearable force sensors may not yet replace force platforms because the attachment of these sensors under the foot or inside footwear could affect the analogue output signal, thus undermining the reliability and accuracy of the force signal. Therefore, simple footswitches are preferred for IC and FO detection if force information is not critical. Footswitches are also chosen to provide the ground truth of IC and FO events in this study. There are two major limitations to this type of sensor. Firstly, they are prone to mechanical failure and have poor durability [55] due to physical wear and tear. Secondly, the sensor is limited in performance in detecting sub-phases of the stance phase and swing phase. These sub-phases are described by a combination of body kinematics and muscle activities in literature. Kinetics data alone does not identify these phases accurately.

Over the past few decades, inertial measurement units (IMUs) are picking up interest in the field of gait phase detection [56]. IMUs are measurement modules that can track its motion in spatial coordinates. This technology is developed from the combination of gyroscope and accelerometer. Chapter 3 would explain the working principle and operation of an IMU in greater detail. IMUs offer the most viable technology to be incorporated with a wearable assistive device because they are portable, durable and relatively inexpensive [34, 54, 55]. This system provides the kinematics information needed to describe the gait phases that would otherwise be difficult to be determined by footswitch/force sensors alone.

The increasing interest in inertial measurement reflects the advancement of sensor technology. MEMS technology allows inertial measurement devices such as gyroscopes and accelerometers to be miniature devices, which are traditionally much bigger in size. Advanced data fusion techniques were developed, so the disadvantage of each type of sensor can be compensated by the other. It is the quality increase in the manufacturing and digital processing that allows IMU to be a viable kinematics measurement and is widely used on devices such as UAVs. Recently, the OpenSense project led by Scott Delp is aimed to use IMU data to interpret human locomotion instead of the traditional markers data from optical tracking. However. IMU measurement is yet to be proven in the laboratory in its ability to extract body kinematics on the same level as the current standard of using 3D motion capturing systems. This study would also use IMU as the primary measurement as it is much more durable and suitable for wearable devices than all other options. The study would also be one of the first examples of using IMUs to interpret body and joint kinematics related to the definition of the gait phase according to established biomechanics theory.

2.1.2. Existing IMU-based GPD Algorithms

Traditionally, a human assessor is required to conduct the detection and classification of gait phases in gait analysis. The labelling could be done semi-automatically by some commercial biomechanics software such as Anybody™ or NEXUS with assessor approval or manually by observing the data and finding the moment it matches the description literature.

A variety of signal processing techniques for gait phase detection have been used and tested. They could be classified as rule-based approaches, machine learning, and hybrid approaches. For rule-based approach, different method of thresholds is applied. They includes fixed-value thresholds to the measurements [54-63], adaptive thresholds on the measurements [64-66]. Thresholds could also be applied to transformed data such as translated symbol [67], frequency or wavelet analysis [37, 68, 69] [70-73]. These rule-based techniques are common because they are computationally efficient and easy to implement on simple electronics.

Machine learning approaches are increasingly popular in the past decade. It includes but not limited to the deployment of machine learning classifier [74-78], Hidden Markov model

[79-84], fuzzy logic [58, 85-87], and neural networks techniques [88-94]. This approach is generally deployed when some of the sub-phase is difficult or ambiguous to define using rules. Implementation of machine learning techniques is commonly offline. Online applications required the exportation of trained models on compatible electronics.

Although many algorithms have been identified in the literature, only several of these methods have translated into real-time implementation [55, 58, 65, 83, 90, 95]. Online implementation of other algorithms has the potential to introduce hundreds of milliseconds of delays due to the large sampling size in their data processing [96], complexity [97, 98] or the requirement of training a model [99]. A recent review conducted concurrently with the study has pointed out the limited online application of GPD algorithms [100].

Another gap for GPD is that stair ambulation is not being investigated as extensively as level walking, most studies are investigating level walking either with different sensors, the signal process methods, and the demographics of test subjects. Among the existing stair GPD found in literature, a smaller number of them have attempt to implement real-time detection. Real-time implementation may also have hundreds of millisecond of timing delay [101].

This study aims to deliver a validated detection algorithm for stair ambulation in real-time within a biomechanical significant timing error. It will be implemented on a physical wearable device and tested on human subjects during stair ambulation.

Prior studies on the topics of gait phase detection have different approach in evaluating the performance. Some previously listed studies reported the successfulness of detection (recall, precision and F1-score), and some other has reported the timeliness (timing error between detection and ground truth) of each detection. Some other studies have other performance indexes unrelated to the gait phases, because they are using the existence of these phases for another purpose. For example, the detection order of the gait events/phases to determine the type of gait activity [76, 97, 102], or using self-defined phases to determine gait parameters [57, 68, 79].

Table 2.1 summarises the timing errors and the recall associated with studies using inertial measurements on wearable devices. Studies which did not report of the timing error are not included in the table. IMU sensor type is defined to be using both the accelerometer and the gyroscope in their detection algorithms.

Table 2.1[1]: Summary of GPD performance in the literature using inertial measurement

Researches	Sensor type	Year	Gait Event	Timing Error	Recall	Activities
Hanlon & Anderson	Insole sensor & Accelerometer	2009	IC FO	2.4 [2.1] 9.5 [9]	NA	Level Walking
Gonza'lez *et al.*	Accelerometer	2010	IC FO IC^ FO^	13 [35] 9 [54] 117 [39] 34 [72]	NA	Level Walking
Aung *et al.*	Accelerometer	2013	IC FO	15.7 [28.6] 5.1 [12.1]	86%	Level and Ramp
Sant'Anna & Wickstrom	Accelerometer	2010	IC FO	50 [40] 30 [40]	NA	Level Walking
Flood *et al.*	Accelerometer	2019	IC FO	<18 <39	NA	Level, inclined, treadmill walking
Catalfmao *et al.*	Gyroscope	2010	IC FO IC FO IC FO	8 [9] -50 [14] 21 [15] -43 [10] 9 [20] -73 [12]	overall 98%	Level Walking Ramp Ascent Ramp Descent
Formento *et al.*	Gyroscope	2014	IC FO IC FO	-11 [18] 35 [20] 18 [46] 132 [44]	93%	Stair ascent Stair descent
Pappas *et al.*	Gyroscope	2001	IC FF HR FO	70 70 40 35	96% (stair) 99% (level)	Stair Ascent Stair Descent Level Walking
Bejarano *et al.*	IMU	2015	IC^ FO^	12 [18] <31 [43]> 5 [32] <7 [55.5]>	0.998 0.944	Level Walking
Maqbool *et al*.	IMU	2017	IC^ FO^ IC^ FO^ IC^ FO^	17 [17.9] <-5.7 [16]> -7.6 [35.2] <-12.8 [6.7]> 14 [21] <-10 [14.7]> -5 [32] <-11.6 [7.6]> 10.5 [17] <-11.8 [16.1]> -25 [36] <-22.8 [10]>	100%	Level Walking Ramp Ascent Ramp Descent
Khan & Biddiss	IMU	2017	IC	250 [200]	96%	Stair Ascent & Descent

[1] Timing errors are represented as the mean [standard deviation]. Negative values indicate the event is earlier than the reference. Timing errors within <> are results from pathological gaits; ^ indicated real-time detection of the gait event.

Kotiadis *et al*.	IMU	2010	IC FO IC FO IC HR FO	NA NA NA NA -10 to -100 [20 to 40] 50 to 130 [20 to 50] NA	77 to 100% 15 to 36%	Stair Ascent Stair Descent Flat walking

This study will use both the successfulness and timeliness of the detection to evaluate the performance. The successfulness of correct classification of each gait phase in terms of recall, precision and F1-score. The timing performance is evaluated with the mean and standard deviation of the timing error between the algorithm detection to the ground truth. The algorithm's usefulness will be verified with the likelihood for a detection exceeding the tolerance of its intended application. To the best of the author's ability, there doesn't have a benchmark of how responsive the detection needed to be. Thus, this study will target a timing error within 50 ms as it is an acceptable range for biomechanical application [36, 37].

2.2. Biomechanics of Stair Ambulation

This section provides background knowledge of stair ambulation's biomechanics and how each gait phase is partitioned and defined in the literature. This section would summarise the key findings from established biomechanics observations.

Table 2.2: Gait partitioning of stair ambulation

Activities	**Phases**				
Stair Ascent	Stance			Swing	
	weight acceptance	pull-up	forward continuance	foot clearance	foot placement
Stair Descent	Stance			Swing	
	weight acceptance	forward continuance	controlled lowering	leg pull-through	foot placement

The stance-swing phase partitioning is common across all lower limb gait. Level walking has five, seven and eight-phase partitioning [103]. Meanwhile, stair ambulation could be partitioned into five phases for ascending and descending gait [104]. The five phases are weight acceptance (WA), pull-up (PU), and forward continuance (FCN) in the stance phase; foot clearance (FC) and foot placement (FP) in the swing phase for stair ascent. Similarly,

weight acceptance (WA), forward continuance (FCN), and controlled lowering in stance phase; leg pull-through (LP) and foot placement (FP) in swing phase for stair descent. Table 2.2 shows the two-phase and five-phase partitioning for both directions of stair ambulation.

Weight acceptance in stair ascent begins with contact with the ground by the swinging leg. During this phase, the body is shifting its weight from the contralateral leg to the ipsilateral leg. The positioning of the weight transfer is achieved by plantarflexion of the ankle. The pull-up phase occurs when the contralateral leg is taking off. The pull-up phase contributes to the most significant progression upward with the knee joint being the major contributor, which is achieved by extending the entire leg. There is no clear transition between the pull-up and forward continuance phase. During the forward continuance phase, the body's movement is progressing forward; however, the progression forward is not separated from the progression upward. The beginning of this phase is often approximated with the mid-swing of the contralateral leg. If a boundary is to be defined, it would be the end of the extension of the ipsilateral leg, and the muscle is either in an isometric or eccentric state just prior to foot-off. Foot clearance is initiated by ipsilateral foot off. During this phase, the leg has to lift and place the foot over the next landing step. The motion is controlled by a series of flexion of the hip, knee and ankle joints. Foot placement begins with the extension of the knee joint during mid-swing. It is when the body has positioned for the landing of the swing leg. The contact of the ipsilateral leg marks the end foot placement.

Weight acceptance in stair descent is defined similarly with foot contact of the ipsilateral leg. During this phase, the knee and ankle muscles absorb the kinetic energy from dropping from the previous step. The moment of the knee and ankle joint is in the opposite direction as their movement. It is regarded as negative power in biomechanics. The phase is then transited into 'forward continuance' when the contralateral foot-off occurs. During this phase, the body is shifting forward. The body would also rise slightly as the contralateral foot takes off. Controlled lowering begins when the body is dropping downward. During this phase, the knee joint is flexing with quadriceps extensors active. The next phase begins with the hip pulls the leg off the step when foot-off occurs. The hip and the knee would continue to flex during early swing. Since the elevated position of the foot on the previous step, the knee joint would only flex slightly while the hip swings the leg forward to the next step during this phase. Foot placement begins with the extension of all three joints during mid-swing. The extension is to place the foot onto the next step. During this phase, the leg will prepare for shock absorption and weight acceptance of the next step.

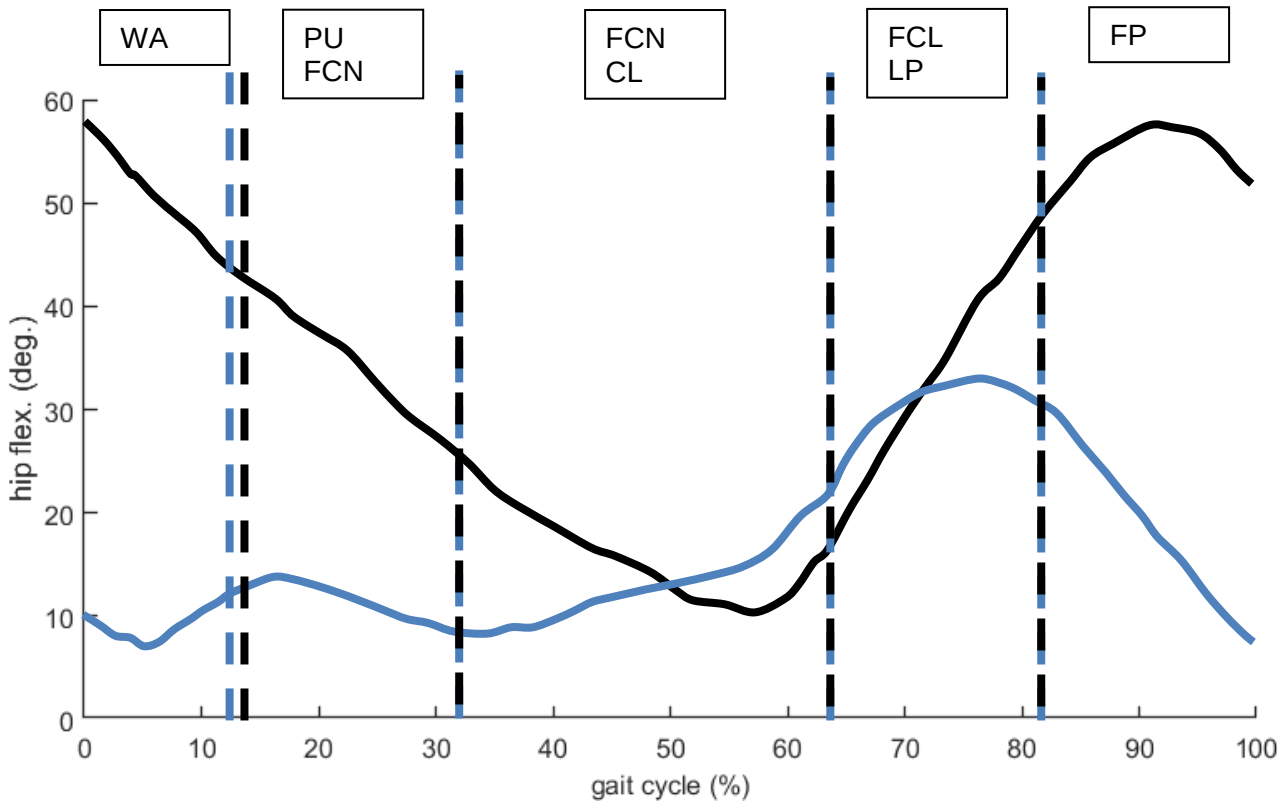


Figure 2.1: The average hip flexion angle during one gait cycle of stair ascent (black) and descent (blue) across 8 subjects. Vertical lines are the transition between the gait phases. Phase codes are defined in the text. Diagram is redraw from data provided in [104]

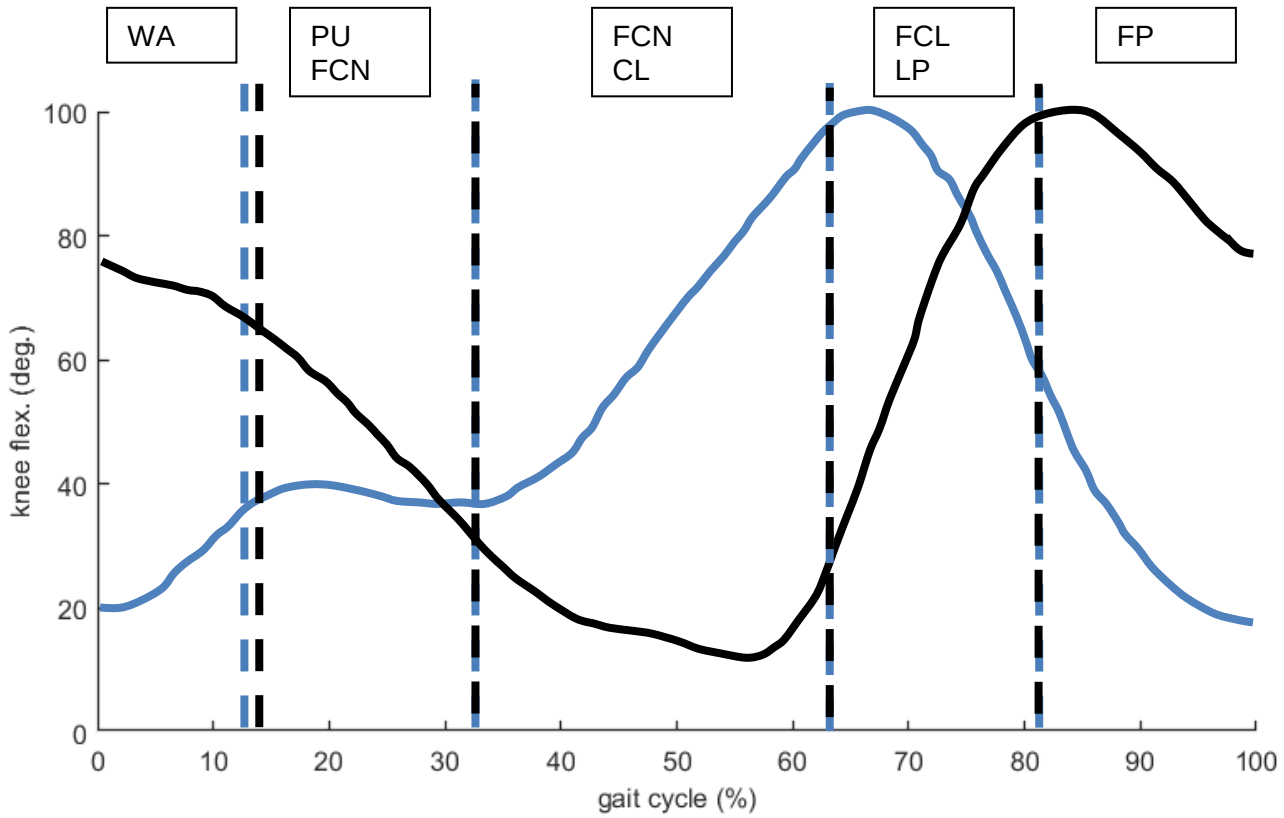


Figure 2.2: The average knee flexion angle of one gait cycle of stair ascent (black) and descent (blue) across 8 subjects. Vertical lines are the transition between the gait phases. Phase codes are defined in the text. Diagram is redraw from data provided in [104]

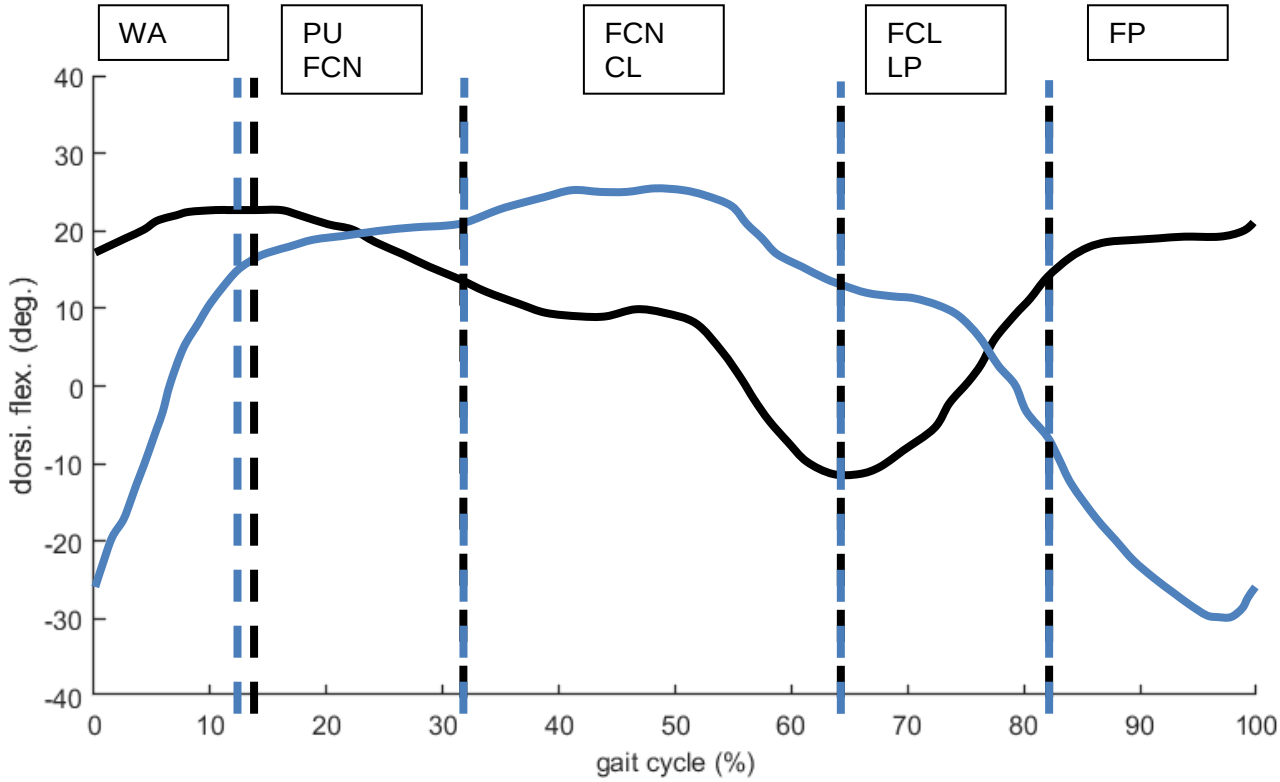


Figure 2.3: The average ankle dorsiflexion angle of one gait cycle of stair ascent (black) and descent (blue) across 8 subjects. Vertical lines are the transition between the gait phases. Phase codes are defined in the text. Diagram is redraw from data provided in [104]

Later studies have added and refined details on the kinematics and muscle activities of the gait [105-108]. Despite the added knowledge, the definition of these five phases remains consistent and widely used. They also expand the observation to gaits in different activities condition [109-113], and other groups of people [114-116].

2.3. Gait Analysis

2.3.1. Measurement Systems

In biomechanics studies, the primary goal is to reveal as much information about the body as possible from the measurement. Therefore, the kinematics and the kinetics of the bodies is essential. These two sources of information are the core of all modern gait analysis.

Kinematics data are typically captured using a vision-based system. Traditionally, human movements are described qualitatively, first by Borelli in 1680. Stereophotogrammetry techniques are the first to quantify the human movement (Braune and Fischer 1895). The

3D reconstruction of the instantaneous position of the bodies in a laboratory is the basis of all 3D motion capturing developed after that. There are several other techniques introduced to measure the body movement such as stereosonics systems [117], exoskeletal linkage with electrogoniometer [118] and accelerographs methods using multiple accelerometers attached to the body [119]. The state-of-the-art technology commonly used in modern gait analysis facilities is 3D motion capturing system using optical markers tracked by multiple cameras. This technology is developed from the original stereophotogrammetry approach with the standardised procedure and added features to improve the system's performance.

The only force that is measurable directly is the ground reaction forces. Force sensors are placed underneath the floor, which the subject would be walking on it. The internal forces of the body and jointed must be computed through the external forces and the kinematics information using a biomechanical model, which the procedure of analysis is explained in section 2.3.2.

For studies that would like to have a better estimation of the internal forces, additional sensors such as EMG are used to provide the muscle activation level for the musculoskeletal model. Although the number of possible sensing is limited compared to the actual number of acting muscles in the body, it does provide some boundary and information for the musculoskeletal model and represent a more accurate image of what truly happened. Other types of equipment such as MRI or X-ray scan provide biological information of the subject that could be used to refine the model as close to the real-life counterpart as possible.

All information gathered from the aforementioned sensors can then imported into biomechanics analysis software for in-depth study.

2.3.2. Analysis Methods

A musculoskeletal model is the centrepiece of modern biomechanical analysis. The kinematics of the markers attached to bodies are used collectively to interpret each joint's movement and the body part kinematics in 3D space. The setup of the markers on the body and the consequence operation to define each joint and joint kinematics are investigated and standardised over the decades [120-123]. The outcomes of these studies are later adapted into commercial systems such as VICON, which has established and verified

procedure for analysis [124]. The validation of the standardised gait analysis allows reproducibility of study if the model is defined consistently.

A pivotal moment in the study is the standardisation of the joint definition published by the International Society of Biomechanics [125, 126]. It allows a general musculoskeletal model to be constructed so that the results of gait analysis are much more repeatable and consistent across the field. From that onward, studies focus on optimising models and experimental protocol that allow more accurate interpretation of the joint toward the ISB's definition and producing more consistent kinematics capturing on the models [127-138]. The technology of 3D motion capturing itself are also being refined [139-142].

Regardless of the experimental protocol and the model being used, the procedure of the biomechanical analysis remains the same. Firstly, static data is capture form the subject so that the model can be scaled to the subject in three dimensions. It is done to make sure the kinematics of the movement reflect those of the subjects. The weight of the subject is also recorded so that the contribution in joint load from the gravity can be separated from those by the muscle when analysing the dynamics of the model.

When analysing the body kinematics using the musculoskeletal model, the software would cluster groups of markers and then define a rigid body with each cluster. Each rigid body's overall movement in 3D space is calculated using the least square fit method of all respective markers within each group. The joint movement is then calculated so that the body defined by each cluster of the experiment data is the least squared fit of those on the model. The resultant kinematics of all joints on the body is one of the main outcomes from the experimental data. The motion data is then used in other analysis processes such as body dynamics, joint loads, and muscle activation/forces. It is important to know that the result of each stage (kinematics, dynamics, joint kinetics, and muscular information) is computed by solving a least-squares optimisation problem.

Due to the underactuated nature of the musculoskeletal model, the results should be checked after each stage that they are following the best practices, and no apparent mistake is found. For example, in OpenSim, an open-sourced biomechanical simulation software that is actively updated by biomechanics researchers, each process is validated by corresponding studies in publications [143-148]. It also has documentation for the operation, verification and validation of models and simulation results [149].

The gait phases aforementioned in section 2.2 are described by the joint movement, the joint loads, and the muscle activation. The definition of gait phases in this study would be as close to the biomechanics definition as possible. Therefore, we will try to interpret the information with the technology available and used in GPD mentioned in section 2.1.

The study will develop a wearable system that can capture the lower limb's kinematics used to interpret the joint angle from the requirement above. The body orientation would be determined by the Cardan angles of each IMU [120, 125]. Thus, the joint angle could be calculated from the angles between two bodies consistent with biomechanics analysis.

Among available wearable technology, footswitches provide a reliable method to establish the ground truth for stance and swing phases, namely the occurrence of initial contact and foot off. A pair of insole footswitches would be used as the ground truth. These two types of sensors would allow the system to capture the two core pieces of information for gait analysis.

The next chapter will provide a comprehensive description of the development and performance of the wearable measurement system that would allow the study to observe and determine the occurrence of gait phases in an out-of-lab environment.

2.4 Research Problems and Aims

To the best knowledge of the author, there are little readily-available proven real-time GPD methods of stair ambulation verified in a real-world environment. The goal of the study is to develop and verify a real-time GPD on a wearable lower-limb device.

The overall challenge could be broken into the following parts:

1. development and evaluation of body sensing measurement on a wearable knee brace that captures critical information
2. transferring the biomechanics knowledge of stairs ambulation into detection rules
3. detailed implementation of a detection algorithm on a minimalistic wearable device
4. development and verification of the real-time gait phase detection algorithm of stair ambulation

CHAPTER 3 : SYSTEM DESCRIPTION

Overview

This chapter is focused on the developed wearable ambulatory system. It provides insights into the technical considerations and key knowledge for developing such systems. The system is designed to be integrated into a knee brace. It gathers crucial and relevant body measurements used in developing and testing GPD algorithms in this research project.

There are two major aspects in the development of a wearable ambulatory system for gait motion study. The IMUs gather repeatable and accurate inertial measurement of their attached bodies. Then, the system integration allows the validation of the algorithm against the ground truth from a pair of the insole sensor, a common reference system used in literature.

This chapter begins with the description of the overall system design and how it fulfils its requirement as a platform for validating GPD algorithms in its relevant application, then the operation of the system and how it can be used. Then, a later section is dedicated to explaining the function and operation of the IMU sensor. This section begins with the theory and working principle of a 6-axis IMU sensor, it is followed by the implementation of the modules in this research project and a performance evaluation of the sensor in terms of repeatability and accuracy against two other methods widely used as the ground truth in literature as mentioned in Chapter 2.3.1.

3.1 Wearable Ambulatory System

3.1.1 System Design

A knee brace was modified to incorporate IMUs (MPU6050) on the shank and thigh segment to measure these segments' sagittal-plane motion and provide the data as input to the microcontroller (PSOC5). The IMUs were mounted on the brace with the local Z axes aligned with the proximal-distal axis of the thigh and the shank segment of the brace. The local X-axes were aligned with the medial-lateral axis.

Two insole footswitches (B&L Engineering, USA) were integrated to identify the reference gait events IC and FO. These footswitches were selected because of their usage in similar research [35] and their acceptance in gait analysis facilities [41]. The switches have their Digital to Analog Converter (DAC) with an R-2R ladder to convert the four contact switches' binary signal to an encoded 4-bits variable as an output with 16 different voltage level, each representing a combination of on-contact.

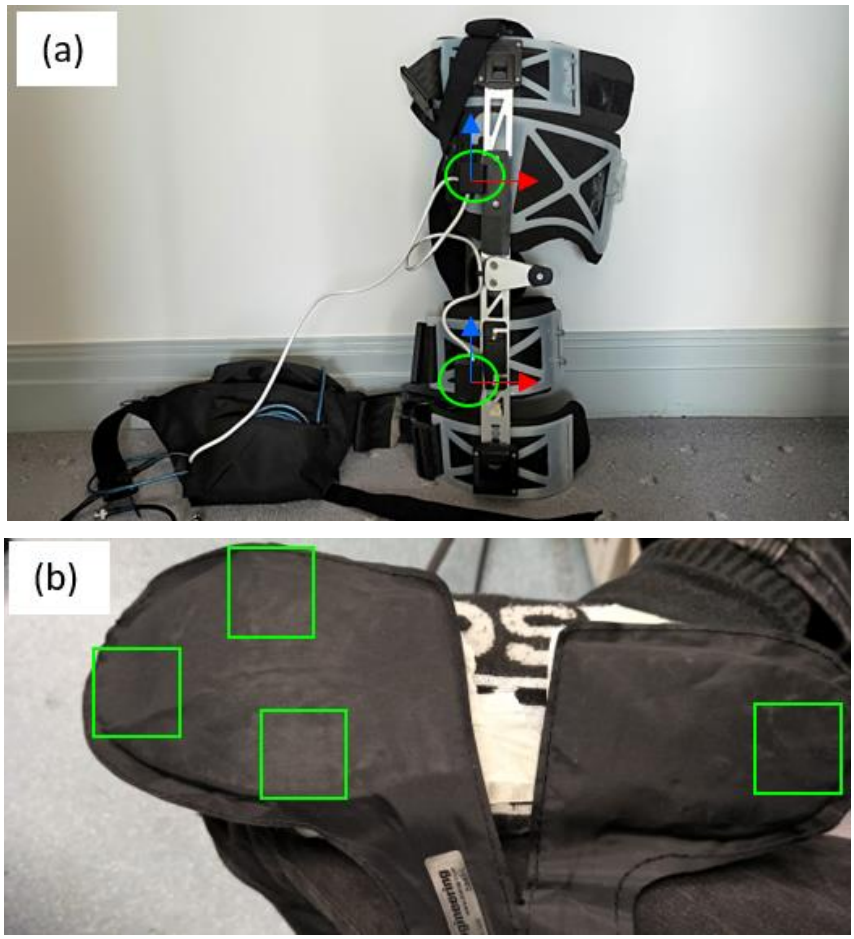


Figure 3.1: Measurement system: (a) the knee brace with IMU attached to the centre of each green circle, red: Y axes, blue: Z axes; (b) insole footswitches, green square indicates the location of each contact area.

Both the IMUs and insole footswitches are then integrated to channel their output to the same microcontroller for data acquisitions. All data is processed in the microcontroller then transfer to the PC via a USB cable which guarantee connection stability and speed.

The system is powered by a battery in the waist bag worn by the user where it stored all electronics components.

3.2 IMU Background

The IMU chosen for this work is InvenSense MPU6050. They are commercially available with low cost, low power and relatively high performance, which are desirable qualities for a wearable device.

MPU6050 houses an onboard Digital Motion Processor (DMP™) that computes motion fusion algorithms from the 3-axis gyroscope and 3-axis accelerometer data via the I2C bus at 400 kHz. It also allows an auxiliary magnetometer to be integrated via the I2C bus. Since the final device will accommodate a motor-driven actuation, the electro-magnetic pulse generated from the motor may cause additional noise to the magnetometer, and shielding will add undesirable weight to the design; therefore, a decision is made not to incorporate a magnetometer in the design.

IMPU6050 allows users to configure the full-scale range of the sensors, the setting of the digital low-pass filter for the sensors, and the output sample rate of the DMP. The full-scale range can be programmed to ±250, ±500, ±1000, or ±2000 degrees per second (dps) and ±2g, ±4g, ±8g, or ±16g for the gyroscope and accelerometer respectively. This work configured the full-scale range to be ±2000 dps and ±8g, to avoid signal clipping during fast movement of the limb. The gyroscope has a raw output rate of 8 kHz, whereas the accelerometer is 1 kHz. Any sampling rate above 1 kHz will cause the repeated output from the accelerometer. Table 3.1 below shows the available configuration of the DLPF and the sensor output rate, and delays.

Table 3.1: Options for configuration of data rate and digital low pass filter of MPU6050

DLPF_CFG	Accelerometer (Fs = 1kHz)		Gyroscope		
	Bandwidth (Hz)	Delay (ms)	Bandwidth (Hz)	Delay (ms)	Fs (kHz)
0	260	0	256	0.98	8
1	184	2.0	188	1.9	1
2	94	3.0	98	2.8	1
3	44	4.9	42	4.8	1
4	21	8.5	20	8.3	1
5	10	13.8	10	13.4	1
6	5	19.0	5	18.6	1

7	RESERVED	RESERVED	8

DLPF_CFG is a system flag to configure the DLPF. This work configured the DLPF with a bandwidth of 188 Hz and 184 Hz for the gyroscope and the accelerometer respectively, which also synchronise the sensors to have an output rate of 1 kHz to the DMP. The bandwidth allows a peak to be constructed with a peak width greater than 10 ms. It is within the range of operation of other similar gait analysis equipment, force plate (typically 300 Hz), and optical tracking (typically 100 Hz) [150].

DMP includes a sensor fusion algorithm to compute the quaternion of the local IMU frame. MPU6050 indicates the yaw with its Z-axis and roll with its X-axis. The mathematical formulation of the quaternion and the conversion to YPR angles, which is more useful for biomechanics interpretation, will be explained in the next section.

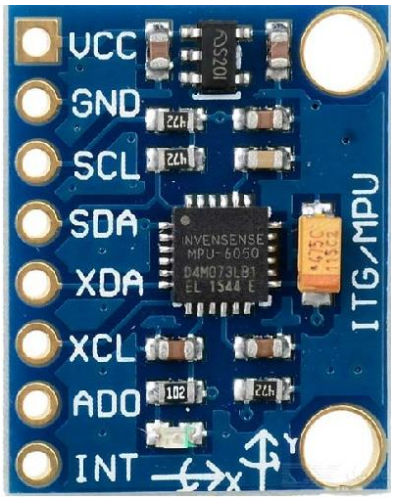


Figure 3.2: Top view of the IMU: X-axis (Roll) points to the right of the page, Y-axis (Pitch) points to the top of the page, Z-axis (Yaw) points out of the page.

3.2.1 Parameterisation of Orientation

There are a few common ways to represent the orientation of the IMU frame: rotational matrix, Euler angles, and quaternion. This section provides the mathematic of each representation and the conversion between them.

1) Rotation matrix

Rotation matrix, $\mathbf{R}$, is a 3-by-3 orthogonal matrix representing the rotational translation of a body from one frame to another frame. It has the property that $\mathbf{R}^{\mathrm{T}} = \mathbf{R}^{-1}$ and $\det \mathbf{R} = 1$.

$$\boldsymbol{x}^0 = \mathbf{R}_1^0 \boldsymbol{x}^1 \tag{3.1}$$

The formulation of a rotational matrix is the sequence of rotation it takes from its initial frame of reference. The combination of multiple rotational is the matrix product of all rotations.

$$\mathbf{R}_\mathbf{n}^\mathbf{0} = \mathbf{R}_\mathbf{1}^\mathbf{0}\mathbf{R}_\mathbf{2}^\mathbf{1} \dots \mathbf{R}_{\boldsymbol{n}}^{\boldsymbol{n-1}} \tag{3.2}$$

The advantage of rotational matric is that it uniquely describes the orientation of the body, which means the nine values in the matrix give a unique orientation of the body in 3D space. This method is not intuitive, since it does not provide any order of the rotation on the axes unless a specific order of Euler angles is defined, and the inverse kinematics is computed from that definition.

2) Euler angles

Any rotation can be defined by a sequence of rotation around three axes known as Euler angles. The full list of convention can be found in Appendix A. This parameterisation is intuitive, and it translates well into the convention used by biomechanics to define the joint angle, and body kinematics [151, 152].

In this study, we would favour the use of Yaw-Pitch-Roll notation to interpret the limb orientation. It is one of the common notations used with the IMU technology for application such as UAV. This convention describes the following order of intrinsic rotation: first rotates the body around its z-axis, then its y-axis, and finally the around its x-axis.

$$\begin{aligned}
&\mathbf{R}(\psi,\theta,\phi) = \mathbf{R}_\mathbf{z}(\psi)\mathbf{R}_\mathbf{y}(\theta)\mathbf{R}_\mathbf{x}(\phi) \\
&= \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi) & \cos(\phi) \end{bmatrix} \\
&= \begin{bmatrix} \cos(\psi)\cos(\theta) & \cos(\psi)\sin(\theta)\sin(\phi) - \sin(\psi)\cos(\phi) & \cos(\psi)\sin(\theta)\cos(\phi) + \sin(\psi)\sin(\phi) \\ \cos(\theta)\sin(\psi) & \sin(\psi)\sin(\theta)\sin(\phi) + \cos(\psi)\cos(\phi) & \sin(\psi)\sin(\theta)\cos(\phi) - \cos(\psi)\sin(\phi) \\ -\sin(\theta) & \cos(\theta)\sin(\phi) & \cos(\theta)\cos(\phi) \end{bmatrix}
\end{aligned} \tag{3.3}$$

However, Euler angles do not uniquely describe the orientation of the body. Firstly, the angle would wrap from each full rotation, so tracking an ongoing spin around an axis would require

external conditions to record and carry over the accumulated angle of rotation. The other problem is gimbal lock; this is when an axis has a rotation of 90 degrees, and the matrix cannot solve for a unique rotation. Let say $\theta = \frac{\pi}{2}$.

$$\mathbf{R}\left(\psi, \frac{\pi}{2}, \phi\right) = \begin{bmatrix} 0 & \cos(\psi)\sin(\phi) - \sin(\psi)\cos(\phi) & \cos(\psi)\cos(\phi) + \sin(\psi)\sin(\phi) \\ 0 & \sin(\psi)\sin(\phi) + \cos(\psi)\cos(\phi) & \sin(\psi)\cos(\phi) - \cos(\psi)\sin(\phi) \\ -1 & 0 & 0 \end{bmatrix}$$
$$= \begin{bmatrix} 0 & \sin(\phi - \psi) & \cos(\phi - \psi) \\ 0 & \cos(\phi - \psi) & -\sin(\phi - \psi) \\ -1 & 0 & 0 \end{bmatrix} \tag{3.4}$$

Hence, any rotation around either z-axis or x-axis, there are alternative solutions. As a result, the YPR angles must be calculated from a more robust form such as rotational matrix or quaternion.

3) Quaternion

This is the preferred method of storing orientation data in this study. This is the most common processed output of any commercial IMU with the internal processor for motion tracking. It does not suffer from gimbal lock and is advantageous over the Euler angles. It also has an advantage over rotational matrix for electronics communication, since it only consists of four number instead of the nine required for transition.

Quaternion is a number system that consists of a scalar (the real number: q_0) and a vector part (the imaginary part: $\mathbf{q_I}$) as described by Hamilton.

$$q = a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k} = q_0 + \mathbf{q_I}, \tag{3.5}$$

with the imaginary part following the rule of

$$\mathbf{i^2} = \mathbf{j^2} = \mathbf{k^2} = \mathbf{ijk} = -1. \tag{3.6}$$

In the application of spatial rotation, quaternion rotation operation is defined as:

$$\mathbf{v}' = q\mathbf{v}q^{-1}, \tag{3.7}$$

where $\mathbf{v}$ is an arbitrary vector ($v_x\mathbf{i} + v_y\mathbf{j} + v_z\mathbf{k}$), and $\mathbf{v}'$ is the vector after the rotation. Since quaternion is a 4D vector, the $\mathbf{v}$ is treated with the real part equal to zero ($0 + v_x\mathbf{i} + v_y\mathbf{j} + v_z\mathbf{k}$) to match the dimension of a quaternion (one real number with three imaginary number).

The inverse of a quaternion is defined:

$$q^{-1} = a - (b\mathbf{i} + c\mathbf{j} + d\mathbf{k}), \tag{3.8}$$

The rotation operation can be interpreted as an encoded form of the axis-angle representation, where a vector, $\mathbf{v}$, rotates θ radian around a unit vector $\mathbf{u} = (u_x, u_y, u_z)$. Then, each coefficient of the unit quaternion is defined as,

$$a = \cos\left(\frac{\theta}{2}\right), b = \sin\left(\frac{\theta}{2}\right)u_x, c = \sin\left(\frac{\theta}{2}\right)u_y, d = \sin\left(\frac{\theta}{2}\right)u_z, \tag{3.9}$$

and

$$q = \cos\left(\frac{\theta}{2}\right) + \mathbf{u}\sin\left(\frac{\theta}{2}\right), \tag{3.10}$$

This definition fulfils the rotation formula of rotating $\mathbf{v}$ around $\mathbf{u}$ by θ radian using the quaternion rotation operation,

$$\mathbf{v}' = \mathbf{v}\cos(\theta) + (\mathbf{u} \times \mathbf{v})\sin(\theta) + \mathbf{u}(\mathbf{u} \cdot \mathbf{v})\big(1 - \cos(\theta)\big), \tag{3.11}$$

3.2.2 Conversion of Parameterisation

The parameterisation of a rotation matrix, Euler angle and quaternion can be converting to each other. The easier way to present the relationship is to convert every parameterisation into the matrix format and related the elements on the matrix between each other.

1) Rotation matrix and Euler angles

The conversion between rotational matrix and Euler angle straightforward. Since we can represent the rotational of each primary axis as a 3-by-3 matrix and the multiplication of each rotational will be equivalent to the rotational matrix representation.

$$\mathbf{R}(\psi, \theta, \phi) = \mathbf{R}_{\mathbf{rot}} \tag{3.12}$$

For the YPR angle, we can construct the rotation matrix by multiply $\mathbf{R_z}(\psi)\mathbf{R_y}(\theta)\mathbf{R_x}(\phi)$,

$$\begin{bmatrix} \cos(\psi)\cos(\theta) & \cos(\psi)\sin(\theta)\sin(\phi) - \sin(\psi)\cos(\phi) & \cos(\psi)\sin(\theta)\cos(\phi) + \sin(\psi)\sin(\phi) \\ \cos(\theta)\sin(\psi) & \sin(\psi)\sin(\theta)\sin(\phi) + \cos(\psi)\cos(\phi) & \sin(\psi)\sin(\theta)\cos(\phi) - \cos(\psi)\sin(\phi) \\ -\sin(\theta) & \cos(\theta)\sin(\phi) & \cos(\theta)\cos(\phi) \end{bmatrix}$$

$$= \begin{bmatrix} \mathrm{r11} & \mathrm{r21} & r31 \\ \mathrm{r12} & \mathrm{r22} & r32 \\ r13 & r23 & \mathrm{r33} \end{bmatrix} \tag{3.13}$$

And the YPR angles(ψ, θ, ϕ) can be calculated from the rotation matrix:

$$\psi = \tan^{-1}\left(\frac{\mathrm{r12}}{r11}\right) \tag{3.14}$$

$$\theta = \tan^{-1}\left(\frac{-\mathrm{r31}}{\sqrt{r32^2 + r33^2}}\right) \tag{3.15}$$

$$\phi = \tan^{-1}\left(\frac{r23}{r33}\right) \tag{3.16}$$

The respective element will be different if a different Euler angle convention is used instead of the YPR angles convention used in this study.

2) Rotation matrix and Quaternions

Similar to how multiplication of complex number can be represented as a multiplication of an equivalent matrix, quaternion multiplication could be present by a 4x4 matrix.

Let p and q are two quaternions, where $p = p_0 + p_1\mathbf{i} + p_2\mathbf{j} + p_3\mathbf{k}$, and $q = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k}$

$$pq = \begin{bmatrix} p_0q_0 - p_1q_1 - p_2q_2 - p_3q_3 \\ p_0q_1 + p_1q_0 + p_2q_3 - p_3q_2 \\ p_0q_2 - p_1q_3 + p_2q_0 + p_3q_1 \\ p_0q_3 + p_1q_2 - p_2q_1 + p_3q_0 \end{bmatrix}$$

$$= \begin{bmatrix} p_0q_0 - \mathbf{p_I} \cdot \mathbf{q_I} \\ p_0\mathbf{q_I} + q_0\mathbf{p_I} + \mathbf{p_I} \times \mathbf{q_I} \end{bmatrix} \tag{3.17}$$

Expand the expression of quaternion rotation, and then group each real and imaginary component we could arrange the matrix form as $\mathbf{R_q}$. Both $\mathbf{v}$ and $\mathbf{v}'$ would have their real component as zero. Full derivation could be found in Appendix B. The solution to the imaginary component (the vector) is also provided in [153-155].

$$\mathbf{v}' = q\mathbf{v}q^{-1}$$

$$\begin{bmatrix} 0 \\ \mathbf{v}' \end{bmatrix} = [\mathbf{R_q}]_{\mathbf{4x4}} \begin{bmatrix} 0 \\ \mathbf{v} \end{bmatrix}$$

$$= \begin{bmatrix} q_0^2 + q_1^2 + q_2^2 + q_3^2 & 0 & 0 & 0 \\ 0 & q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 0 & 2(q_1q_2 + q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 0 & 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \begin{bmatrix} 0 \\ v_x \\ v_y \\ v_z \end{bmatrix} \quad (3.18)$$

Note that the multiplication of $q\mathbf{v}$ would make the real component equal to zero and subsequently with $q\mathbf{v}q^{-1}$, and the dimension of the $\mathbf{v}$ would not change as a result. Therefore, the expression could be written as a 3-by-3 matrix. This resultant matrix is the rotation matrix in term of quaternion components.

$$\mathbf{v}' = [\mathbf{R_q}]_{\mathbf{3x3}}\, \mathbf{v}$$

$$= \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} \quad (3.19)$$

The revert operation could be done by finding the corresponding element on the rotation matrix to produce q_0, q_1, q_2, q_3.

$$q_0 = \frac{\sqrt{1 + r11 + r22 + r33}}{2} \quad (3.20)$$

$$q_1 = \frac{r32 - r23}{4q_0} \quad (3.21)$$

$$q_2 = \frac{r13 - r31}{4q_0} \quad (3.22)$$

$$q_3 = \frac{r21 - r12}{4q_0} \quad (3.23)$$

3) Euler angles and Quaternions

Given that we got the matrix form of both Euler angle and quaternion, we can simply equal the two matrices to get the conversion between YPR angle (ψ, θ, ϕ) and quaternion.

From quaternion to YPR angles:

$$\psi = (\frac{2(q_1 q_2 + q_0 q_3)}{q_0^2 + q_1^2 - q_2^2 - q_3^2}), \tag{3.24}$$

$$\theta = \left(\frac{-2(q_1 q_3 - q_0 q_2)}{\sqrt{2(q_2 q_3 + q_0 q_1)^2 + (q_0^2 - q_1^2 - q_2^2 + q_3^2)^2}} \right), \tag{3.25}$$

$$\phi = (\frac{2(q_2 q_3 + q_0 q_1)}{q_0^2 - q_1^2 - q_2^2 + q_3^2}), \tag{3.26}$$

and, from YPR angles to quaternions:

$$q_0 = \cos\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) + \sin\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right) \tag{3.27}$$

$$q_1 = \sin\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) - \cos\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right) \tag{3.28}$$

$$q_2 = \cos\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) + \sin\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right) \tag{3.29}$$

$$q_3 = \cos\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right) - \sin\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) \tag{3.30}$$

3.2.3. Formulation of the orientation of IMU

This section provides the working principle of IMUs and how the orientation is interpreted from the onboard accelerometer and gyroscope. Here explains the working principle of the IMU and how the orientation information is calculated from the sensor's raw data, which are the linear acceleration and the angular velocity.

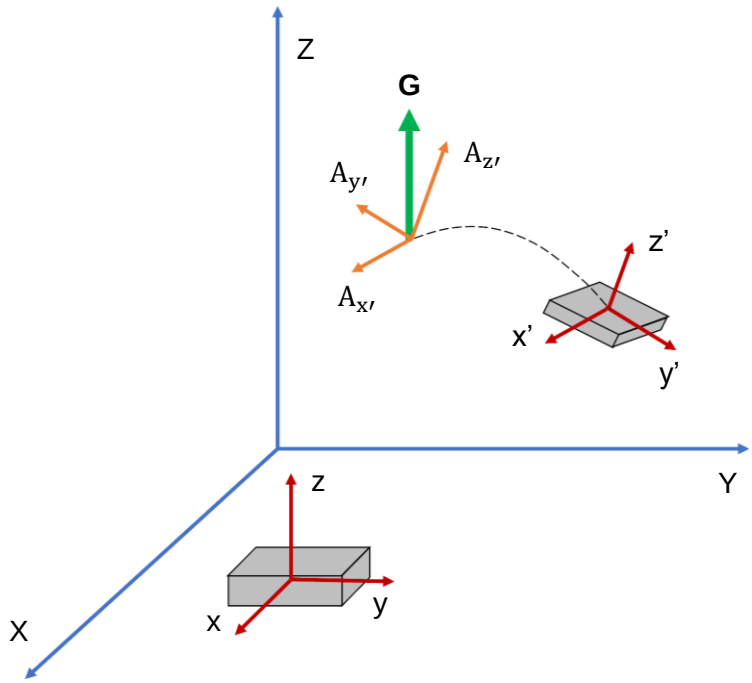


Figure 3.3: A 3D representation of an IMU rotation, Green arrow is the gravity reaction force measured by the IMU, and oranges are the gravity reaction force projection in the IMU transformed frame which are the acceleration measured by the IMU in each axis. XYZ is the world frame, xyz is the IMU original frame, x'y'z' is the IMU transformed frame.

Consider an IMU in 3D space, figure 3.3, it is initially aligned with the world frame. Then it is at a stationary position after some rotation in each axis. The XYZ axes are in the world xyz is the IMU original frame, and x'y'z' is the IMU transformed frame. In a stationary position, the only force acting on the IMU is due to Earth's gravity. IMU measure the reaction to the external force acting on it with its MEMS, hence it would have a reading of 1g pointing upward. Let this reaction force be **G**. $A_{x'}$, $A_{y'}$, and $A_{z'}$ are the acceleration measured by the IMU in its x', y', and z' axis respectively.

The **G** vector projects itself on to the three axes of the accelerometer in reality. Initially the IMU's frame is align with the ground frame, and the **G** vector measured on the IMU is 1g upward. The transformation of the **G** vector to the measured acceleration of the IMU's frame is the revert operation of the transformation of the IMU from the initial frame to the transformed frame. The YPR rotation of the IMU is described by (3.13) in the world frame. Thus, the transformation of the **G** vector in the IMU's frame is the reverse, $\mathbf{R_x}(\phi)\mathbf{R_y}(\theta)\mathbf{R_z}(\psi)$ [156]. ψ is the angle between the Z-axis of the IMU to the z-axis of the world frame, similarly for θ and ϕ for the Y-axis and X-axis respectively.

$$\mathbf{A} = \begin{bmatrix} A_{x'} \\ A_{y'} \\ A_{z'} \end{bmatrix} = \mathbf{R_x}(\phi)\mathbf{R_y}(\theta)\mathbf{R_z}(\psi) \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

$$= \begin{bmatrix} -\sin(\theta) \\ \cos(\theta)\sin(\phi) \\ \cos(\theta)\cos(\phi) \end{bmatrix} \tag{3.31}$$

Note that the negative of the angle is not essential when we reverted the operation, as we are going to solve for both solution within the range of $(-\pi, \pi]$. 3.32 and 3.35 are the general equations to solve for both solutions. 3.33, 3.34 and 3.36, 3.37 are the equations of finding each solution for ϕ_1, ϕ_2, θ_1, and θ_2 respectively. The subscript of 1 and 2 indicate the two solutions from the general equation. n is an integer multiple of π that allow the equation to obtain all solution within $(-\pi, \pi]$.

$$roll : \phi = \tan^{-1}\left(\frac{A_{y\prime}}{A_{z\prime}}\right) + \pi n, \text{ for } \phi \in (-\pi, \pi] \tag{3.32}$$

$$\phi_1 = \tan^{-1}\left(\frac{A_{y\prime}}{A_{z\prime}}\right) \tag{3.33}$$

$$\phi_2 = \begin{cases} \phi_1 + \pi \text{ , if } \phi_1 \leq 0 \\ \phi_1 - \pi, \text{if } \phi_1 > 0 \end{cases} \tag{3.34}$$

$$pitch : \theta = \tan^{-1}\left(\frac{-A_{x\prime}}{\sqrt{{A_{y\prime}}^2 + {A_{z\prime}}^2}}\right) + \pi n, \text{ for } \theta \in (-\pi, \pi] \tag{3.35}$$

$$\theta_1 = \tan^{-1}\left(\frac{-A_{x\prime}}{\sqrt{{A_{y\prime}}^2 + {A_{z\prime}}^2}}\right) \tag{3.36}$$

$$\theta_2 = \begin{cases} \theta_1 + \pi \text{ , if } \theta_1 \leq 0 \\ \theta_1 - \pi, \text{if } \theta_1 > 0 \end{cases} \tag{3.37}$$

The second solution represents the other viable solution where both axes are flipped. A logical operation is required to determined which solution is the correct one by choosing the solution that is physically closer to its previous position.

The equation shows that the IMU could only determine the tilt angles in any stationary position when it is only affected by the gravity only. This is because the gravity only does not provide enough information to solve the yaw angle. The IMU could be resting at any yaw displacement, and the acceleration measured by the IMU is the same.

Generally, an accelerometer is accurate for a long-term measurement; it has no integration drift, and the gravity on Earth is mostly constant. Thus, it can accurately determine the IMU tilting in pitch and roll angle given there is no acceleration either than gravity. This method

is useful when correcting the roll and pitch angle when the IMU come to a rest, and thus removed the accumulated error from the gyroscope during motion.

Limitation of the accelerometer with known gravity is that it could have any yaw angle solution and it works when the IMU is stationary. It is also vital that a low pass filter is implemented on the accelerometer signals because any vibration can cause the calculation to be unreliable.

Pose estimation [157] has been accomplished using accelerometers alone. They typically include a known model for their application so that the gravity acceleration can be separate from the acceleration from the user's motion.

When the IMU is in motion, it is often more reliable to integrate the rotational velocity from the gyroscope to get the tilt angles. Consider the case of one axis gyroscope below:

$$\theta(t + \Delta t) \approx \theta(t) + \omega' \Delta t + \varepsilon \tag{3.38}$$

$\theta(t)$ is the angle at time t , Δt is the time interval between the current and next time instance. ω' is the current gyroscope angular velocity reading. ε is the error model of the gyroscope.

Gyroscope typically has a model of:

$$\omega' = \omega + b + \eta \tag{3.39}$$

ω is the true angular velocity, b is the bias, η is the Gaussian noise with zero-mean. Bias is a constant offset in reading due to temperature. Therefore, it is a best practice to remove the bias. The bias removal is done by observing the long-term output and applying a constant offset to the gyroscope reading.

IMU tilt angle is typically calculated as a complementary between the solution from gyroscope and accelerometer signals. The simplest form can be described as a ratio between the two types of signal, for example, the roll angle:

$$\theta(t) = \alpha(\theta(t - \Delta t) + \omega' \Delta t) + (1 - \alpha)\left(\frac{A_{y'}}{A_{z'}}\right) \tag{3.40}$$

When stationary, the solution of the gyroscope is close to zero, so the output will be from the solution for the accelerometer, which has no drift and no bias, assuming the sensor is calibrated and filtered to minimise the noise. When in motion, the gyroscope solution will change the tilt angle quickly, where the accelerometer won't be useful due to the additional acceleration.

To compute the orientation information accurately in real-time, many sensors deploy a data fusion algorithm between the gyroscope and accelerometer, so the output has minimised error in the output angle. The next section describes one of the common approaches, Kalman Filter, to illustrate how gyroscope and accelerometer complement each other in their angle outputs.

3.2.4 Data fusion

Kalman filter is a popular data fusion technique used in IMU's angle calculation. The general principle of the Kalman filter is to establish and continuously update a covariance matrix of some unknown variables so that the error between the measurement and the prediction based on the system is minimised. The Kalman filter has been well established for decades [158]. The full state space derivation can be found in Thack and Lacey [159]. The mathematics operation of the Kalman filter in different applications is also mentioned in Kim and Bang [160]. This section only provides an overview of the Kalman filter's operation to avoid redundancy and provide a context to its application in IMU.

The operation of a Kalman filter consists of two stages: prediction and update. The next states and state error covariance matrices are predicted based on the system model, a state transition matrix. Once the new measurement is made, different measures and predictions are made, called residual, and the Kalman gain is computed. Then, the state estimate and the state error covariance are updated with the computed Kalman gain. These updated state and state error covariance will be used in the next prediction. The equations of each operation found in the literature [159] are provided below. Note that the Kalman gain is multiplied with the residual to produce a correction to the state estimate so that it can be viewed as a conversion from the measurement to the state variables.

Table 3.2: Mathematics equations of Kalman filter in each operation

Operation	Equations
Initial estimate	x_{k-1}, P_{k-1}
Prediction	$x_k^- = Ax_{k-1} + Bu_k$ $P_k^- = AP_{k-1}A^T + Q$
Kalman Gain	$K_k = P_k^- H^T (R + HP_k^- H^T)^{-1}$
Update	$x_k = x_k^- + K_k(z_k - Hx_k^-)$ $P_k = (I - K_k H)\, P_k^-$

x_k is the state vector at time k, P_k is the error covariance matric at time k; z_k is the actual measurement. A is the state transitional matrix between time k and time k+1, B is the input transitional matrix, u_k is the input to the system, K_k is the Kalman gain, H is the transitional matrix between measurement and state. Q and R are the covariances of the two noise models for x (the process) and z (the measurement), respectively. Kalman filter is a tool to better estimate the output under the influence of known error. It could be combined with other techniques to enhance the data.

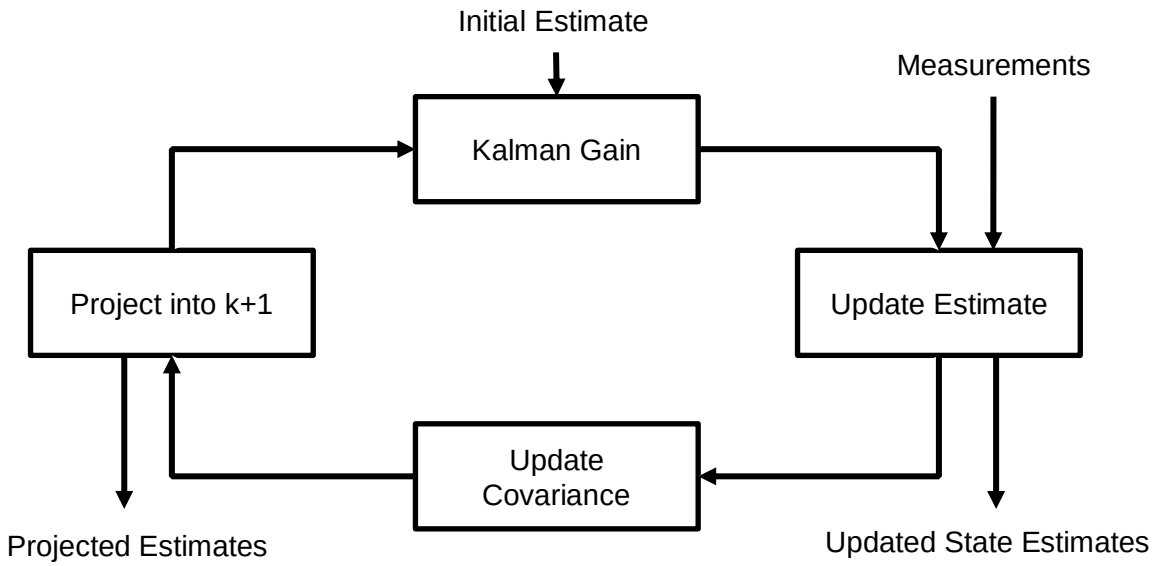


Figure 3.4: Flow chart diagram of a Kalman Filter operation

In more recent studies [161, 162], researchers are developing data fusion techniques to estimate poses (three rotations and three translation) of the IMU. These often include additional vision sensor [163] and a known model of the system [164]. A model allows the prediction of reading between the accelerometer and gyroscope. It can provide a more accurate update on the dynamics. Modelling may not be available in a situation where the sensor placement is unknown. For the knee brace used in this study, the distance between the thigh IMU to the hip joint, and the shank IMU to the foot are unobservable. This distance

would be different for different user. Once again, this study focuses on algorithms to detect gait phases using current measurement technology; the development of more accurate pose estimation of the body is beyond this study's scope. The next section describes the implementation and performance of the IMU in this study.

3.3 IMU Performance

3.3.1 Implementation of the IMU

The implementation of the IMU in this study takes advantage of existing sensor technology and techniques. InvenSense's DMP™ has a proprietary data fusion algorithm for their IMUs, which would compute a quaternion output from its embedded accelerometer and gyroscope. We deploy the manufacturer's recommended practice to prepare and configure the sensor for the DMP™ calculation.

Due to the manufacturing uncertainty and the temperature factor, it is crucial to calibrate the sensor. The output should be as close to 1g in the z-axis as possible when stationary on a flat surface.

A program is made to calibration the IMU in a stationary position on a flat surface at laboratory room temperature. The program will take raw data from the IMU continuously until the offset applied on the sensor's reading gives a stable output within a user-specific accuracy. The offset is then applied to the sensor's onboard processor directly by editing the offset values in its register. This static calibration is aimed to reduce the bias and any mechanics defect the sensor has and make the sensor output exactly 1g in the z-axis. The pseudo-code of the program is below:

```
Read sensor for the first time

While
        Calculate moving average of sample in buffer
        Apply mean as offset to the sensor
        Check if the difference between the reading and the mean is below noise
tolerance
                If so, start counter of valid offset
        If the number of cycle with valid offset is bigger than 10, exit calibration
End
```

The offset required for the thigh IMU is shown below. This offset will remove the constant error from the sensor in each axis, three linear accelerations and three rotational velocities.

```
//Thigh IMU
//Sensor readings with offsets:  5 8 4090 1 1 1
//Your offsets: -2207 -6489 1618 -6 -10 -13
```

In the example above, the calibration has significantly reduced the error caused by the bias and the defect, from 7, 11, 14 bits in raw gyroscope data to 1, 1, 1 bit.

However, integration drift still occurs when the IMU is in motion due to the accumulated error in digital integration. Please refer to section 3.2.3.

It is known that DMP™ use an advanced data fusion technique to compute the IMU's quaternion output to provide a consistent reading. The technique has the ability to update and correct in real-time. It is evident from the response of the IMU's quaternion when converted to YPR angles. The figure 3.5 below shows the computed yaw angle of the shank IMU captured after startup. The signal requires a short amount of time after startup for the output signal to settle.

For a 6-axes IMU, the Yaw angle would be unreferenced and should be drifting without correction. However, DMP seems to have an internal steady-state correction for that as well. This yaw correction could be achieved by modelling the drift over a long time and applying a gradient offset to the reading. The information on the algorithm is proprietary, and it is unclear how exactly DMP calculate its output. Nevertheless, we can still evaluate the performance of an IMU and determine whether this is a suitable sensor.

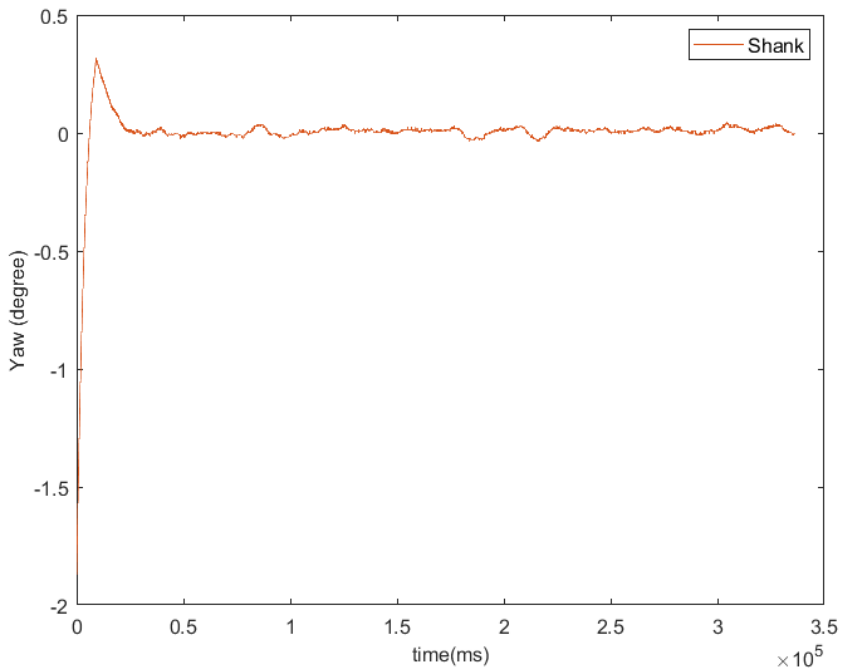


Figure 3.5: Yaw angle of the calibrated Shank IMU including the first 5 minutes after startup

We could consider the sensor output in two operational conditions to evaluate the performance of the IMU in identifying its orientation angles. These conditions are to evaluate the performance in the reliability of the sensor reading during static and dynamic environments.

The next section will describe the experiment conducted to verify the IMU implementation's performance to ensure its output is reliable.

3.3.2 Performance Tests

The IMU measurement is collected after they are integrated with the brace, so the results from this experiment would also verify the performance of the sensor on the knee brace. IMU sensors are calibrated and configured in accordance with the procedures described in section 3.2.3 implementation.

Figure 3.6 shows the experiment setup of the performance test. Actuation unit and quadrature encoder are temporarily installed to verify the IMU performance.

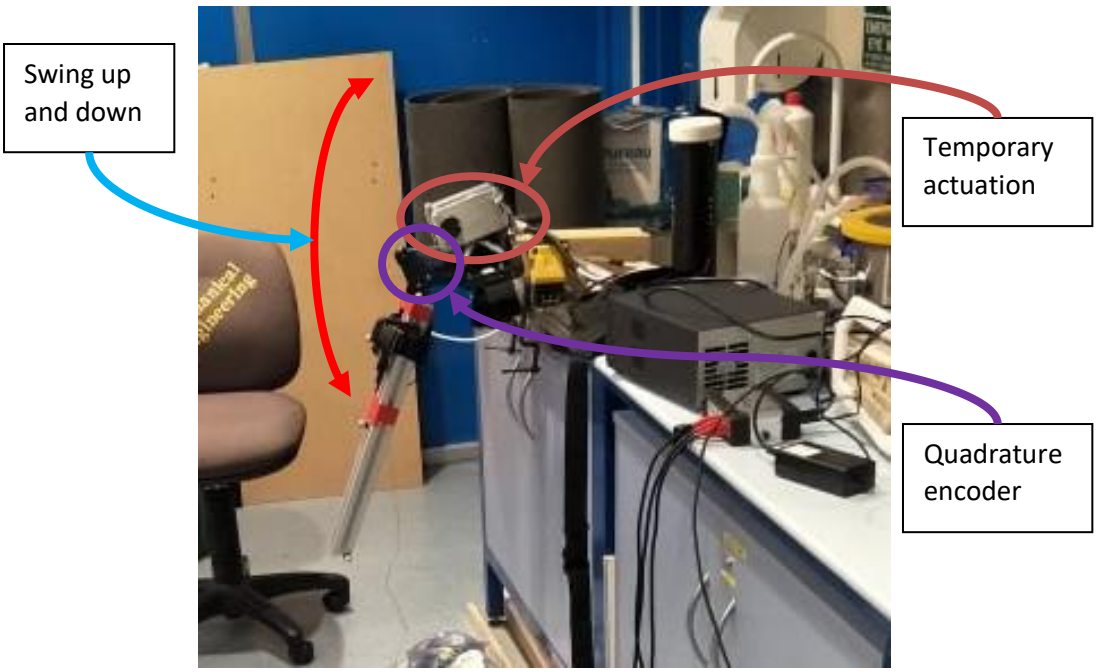


Figure 3.6: Experiment setup of the IMU performance tests. Orange: the location of the actuation which drive the shank segment up and down (the red arrows). Purple: the location of the encoder to measure the angle difference between the brace's segment.

1) Static test:

In this section, we are interested in the IMU measurement noise; therefore, a reference system that indicates the true angle is not required. The study of evaluating the IMU's true angle is explored in the continuous motion test. We can observe the error between the IMU indicated angles to the true angle from an encoder.

The static test was conducted to evaluate the IMU performance in maintaining a stable reading. The brace is secured in a vertical position on a pole. Since we are interested in the IMU measurement noise, a reference system that indicates, the true angle is not required. The study of evaluating the IMU's true angle is explored in the dynamics test. We can observe the error between the IMU indicated angles to the true angle from an encoder. The study assumes both IMUs are identical in performance and the thigh sensor is intentionally uncalibrated to see the data performance before and after the calibration.

Ten trials of 5-minute data are sampled for the statics test. I obtained the steady-state output of the IMU's YPR angles separately. The system reset is done electronically from a distance to minimise the chance of physical contact with the system during and between the trials. The reset is done to make sure the data fusion begins anew, and the output angles are freed from any historical effect of the previous sampling.

Matlab is used to perform all statistical analysis. The data was used for noise analysis to determine the standard error of the sensor's measurement. The noise is the fluctuation in the data, and it is calculated as the difference between two consequent data points. The standard deviation of the measurement is the root mean square of the noise, and it can describe the noise level. The mean standard deviation across all trials is the expected noise of the IMU measurement. The standard deviation of the standard deviation across all trials can describe the level of fluctuation of the noise. The repeatability of the measurement is analysing with the difference in the mean angles across the trials. Figure 3.8 show the standard variation of the static reading of the calibrated IMU over the ten trials. Table 3.3 summarises the statistic of the IMU's YPR angles and their error.

Table 3.3: Summary of the statistical measure of the static trials

	intertrial mean	intertrial SD	mean(intratrial SD)	SD (intratrial SD)
Yaw	0.044104	8.93271E-05	0.000209	1.96779E-05
Pitch	-0.05999	5.66125E-05	0.000192	2.23053E-05
Roll	-0.09649	4.13737E-05	0.000177	3.08353E-05

The performance of the IMU is very accurate and consistent during static conditions. Since the reading of the angle is in degrees. The variation of the output on average is at most 0.0002 degrees in the yaw angle.

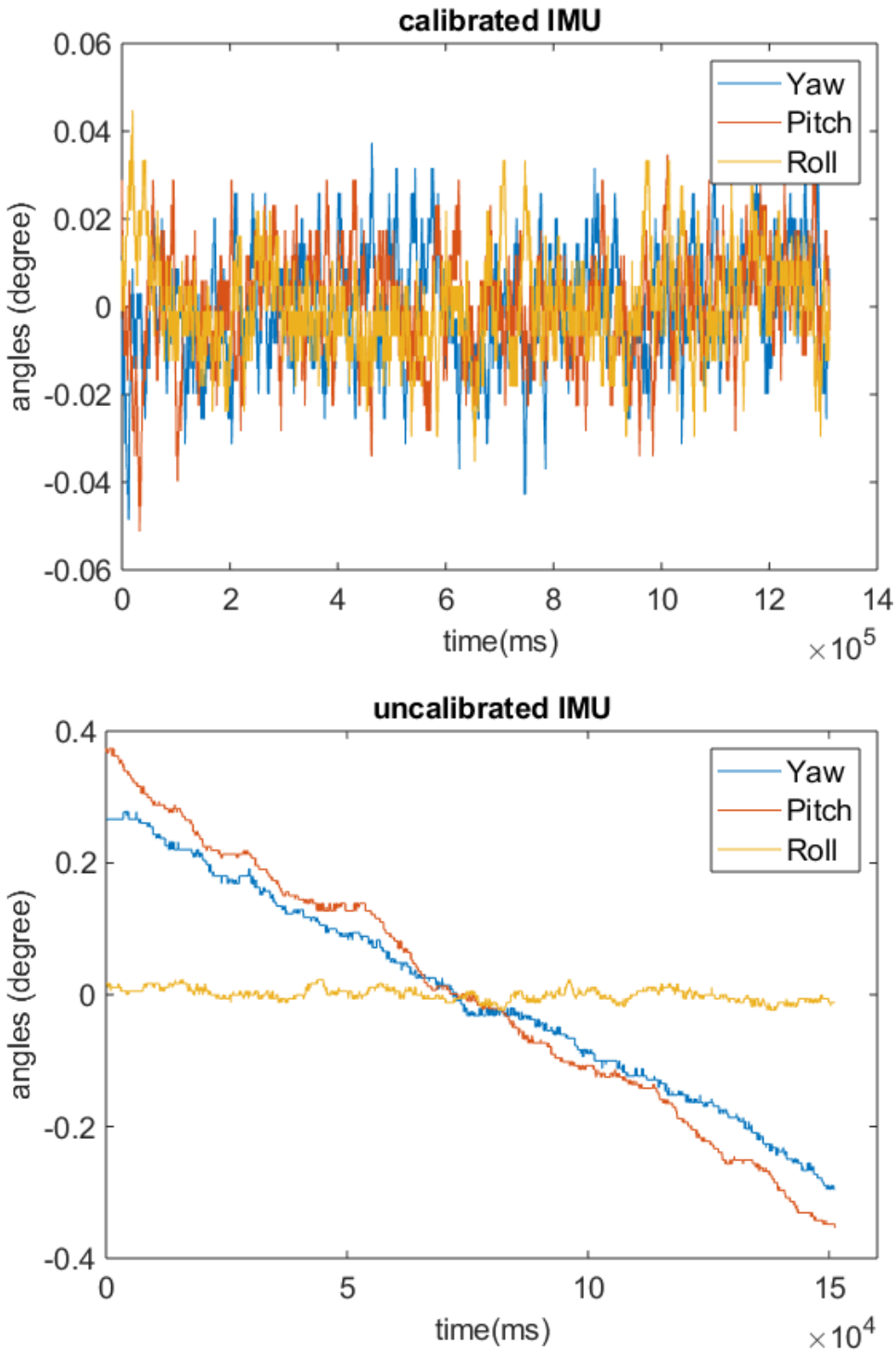


Figure 3.7: Static noise of Yaw-Pitch-Roll angles of the calibrated (top) and the controlled (bottom) IMU

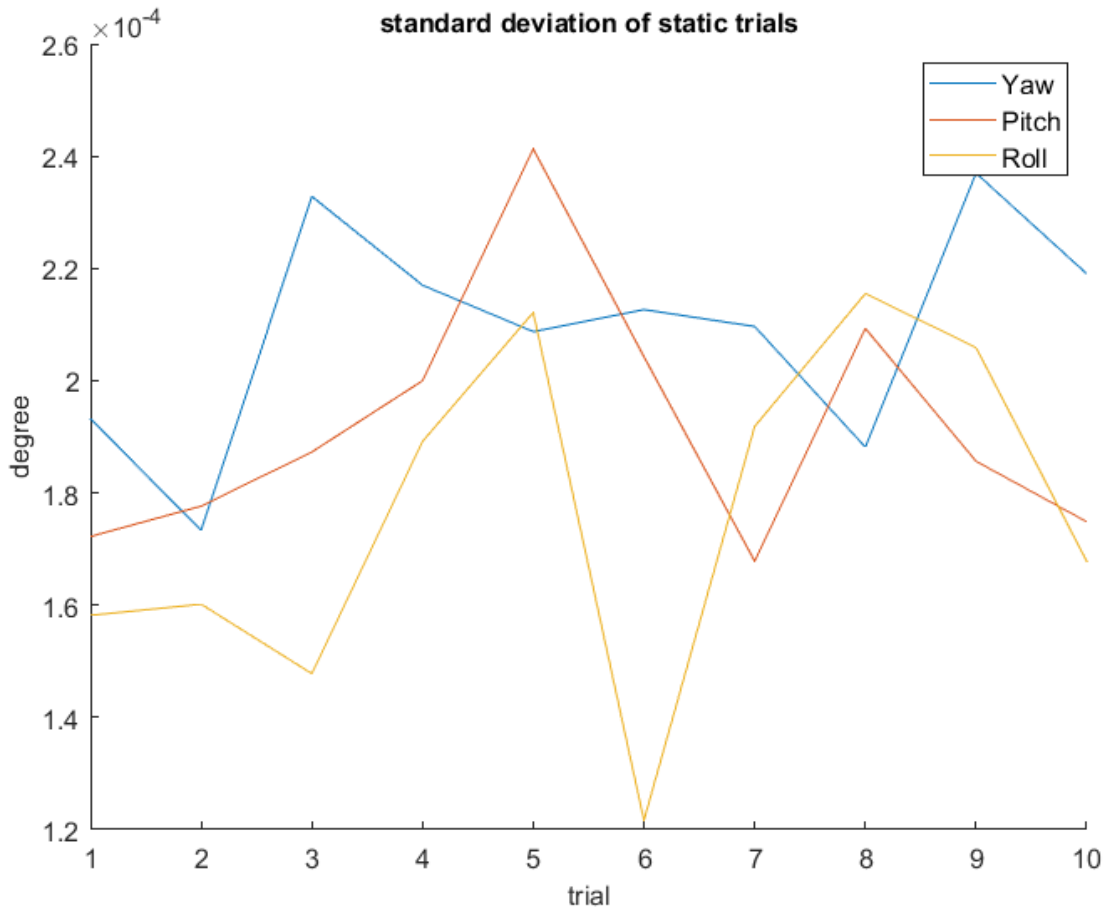


Figure 3.8: Standard deviation of the static trials of the calibrated IMU.

2) Continuous motion test:

The continuous motion test was conducted to evaluate the IMU performance in following the true motion of the attached body. The brace is secured on a bench via the thigh piece. The shank piece is unrestricted, and the knee joint is free to rotate. A quadrature encoder is installed to read the rotation of the knee joint. The actuation is a rotary motor which pulls the shank piece to extend the knee joint, and the gravity will flex the knee joint when the motor reduces its torque output. The purpose of the test is to observe the motion error of the IMU from the encoder angle during continuous motion.

The test is conducted with the following procedure. Firstly, the brace and actuation module are secured on a test bench. Then, a program is run to extend the brace joint for 0.5s, and then switch off the actuation for another 0.5s. The 1s cycle is repeated for the next 5 minutes. These 5 minutes of continuous motion will form one trial of the test. Each trial is repeated ten times at different time interval over a period of 5 hours.

In this test, we will not power cycle the device. Since there is no control over the kinematics of the brace, and we cannot guarantee the IMU would start from the same position, the quick calibration programmed upon startup might introduce more error to the reading.

Since both systems observe the same motion over the duration, the peaks and troughs from the two datasets should match each other. And, we are more interested in the consistency between the two system instead of obtaining an accuracy trajectory of the motion. Matching these points would be equivalent to synchronous the two datasets with time. Results analysis will only use these matching points.

Two-sample Kolmogorow-Smirnow tests (KS test) check if the data between two samples are from the same continuous distribution at the 5% significance level, see code below:

```
>>[ks2hp,ks2pp] = kstest2(enc_peak,IMU_peak)
>>[ks2ht,ks2pt] = kstest2(enc_trough,IMU_trough)
ks2hp =  logical   1
ks2pp =   2.8937e-34
ks2ht =  logical   1
ks2pt =   3.2575e-40
```

The result of the test rejects the null hypothesis that the two datasets belong to the same distribution.

It is possible that there is an offset in their angle measurement. Since we are interested in whether the IMU could track the same motion as the encoder, we can use the KS test after removing the mean of each type of turning points from both datasets. KS test cannot reject that the dataset belongs to the same distribution after offsetting the mean values from each type of turning points between IMU and the encoder, see code below:

```
>>[ks2hp,ks2pp] = kstest2(enc_peak-enc_peak_mean,IMU_peak-IMU_peak_mean)
>>[ks2ht,ks2pt] = kstest2(enc_trough-enc_trough_mean,IMU_trough-
IMU_trough_mean)
ks2hp =  logical   0
ks2pp =    0.9870
ks2ht =  logical   0
ks2pt =    0.9445
```

The above two tests indicate that the two samples have a different mean, but the distribution around their means are very similar to a p-value at 0.9445 and 0.987 for troughs and peaks respectively. It is expected that they shared the same variance but different mean.

Two-sample T-test and F-test are performed to check if each type of turning points shares the same mean or the same variance, respectively.

```
>>[thp,tpp] = ttest2(enc_peak,IMU_peak)
>>[tht,tpt] = ttest2(enc_trough,IMU_trough)
>>[vhp,vpp] = vartest2(enc_peak,IMU_peak)
>>[vht,vpt] = vartest2(enc_trough,IMU_trough)
thp =     1
tpp =   3.9742e-48
tht =     1
tpt =   3.8369e-74
vhp =     0
vpp =    0.4023
vht =     0
vpt =    0.9157
```

T-tests reveal the two systems have a different mean, and the F-tests reveal the two systems are from a normal distribution with the same variance.

The same analysis is performed on all 5-minute samples over the 5 hours duration. Table 3.4 summaries their statistics and the tests.

Table 3.4: Intertrial statistical results

	IMU	**Encoder**	**Δ mean**
Peak	88.5036 [2.8457]	82.3059 [2.6183]	6.1977
Troughs	31.3592 [1.5891]	25.7046 [1.6396]	5.65458
Average			5.9261

The continuous test result indicates a mean difference between the two systems of about 6 degrees, with peak having a greater difference of about 6.2 degrees, whereas the troughs are about 5.65 degrees. Figure 3.9 and 3.10 shows the mean and standard deviation between the system on each event has an apparent separation.

Table 3.5: Results of continuous test between the IMU and the reference encoder system.

	Start	IMU	Encoder	Δ mean	KS-test		T-test		F-test	
1	0:00	87.6127	81.8185	5.7942	1	2.8937e-	1	3.9742e-	0	0.4023
		[2.0378]	[1.8659]			34		48		
		30.0556	24.8778	5.1778	1	3.2575e-	1	3.8369e-	0	0.9024
		[1.1332]	[1.1458]			40		74		
2	0:22	89.3719	83.1061	6.2658	1	1.4849e-	1	6.2572e-	0	0.1944
		[2.3499]	[2.0655]			33		51		
		31.0122	24.9067	6.1055	1	1.6021e-	1	1.0551e-	0	0.9977
		[1.2968]	[1.2965]			45		84		
3	0:47	88.7948	82.6383	6.1565	1	4.0233e-	1	4.1825e-	0	0.3300
		[2.9571]	[2.6856]			26		37		
		30.9848	25.4604	5.5244	1	1.0129e-	1	1.6719e-	0	0.9697
		[1.6843]	[1.6907]			38		60		
4	1:52	90.9475	84.5185	6.4290	1	5.5918e-	1	3.0109e-	0	0.4108
		[2.9971]	[2.7940]			34		50		
		30.7323	24.9469	5.7854	1	6.7989e-	1	2.4848e-	0	0.6036
		[1.6048]	[1.6778]			58		86		
5	2:13	90.5600	84.1730	6.3870	1	2.7792e-	1	2.6079e-	0	0.4391
		[2.7463]	[2.5441]			30		42		
		31.2198	25.4337	5.7861	1	4.2275e-	1	2.3063e-	0	0.7167
		[1.7507]	[1.8150]			39		59		
6	2:34	90.7070	84.2202	6.4868	1	3.9888e-	1	2.2750e-	0	0.5312
		[2.4071]	[2.2685]			37		54		
		31.2152	25.3623	5.8529	1	1.3065e-	1	3.2706e-	0	0.8203
		[1.4543]	[1.4859]			48		80		
7	2:59	89.1888	82.8488	6.3400	1	1.2851e-	1	2.1824e-	0	0.5472
		[3.3251]	[3.1424]			23		35		
		31.7222	26.0210	5.7012	1	4.0212e-	1	9.0343e-	0	0.4692
		[1.8297]	[1.9584]			41		61		
8	3:39	88.4079	82.1304	6.2775	1	2.0190e-	1	4.7957e-	0	0.5158
		[2.3069]	[2.1773]			44		62		
		31.6263	25.9625	5.6638	1	3.2357e-	1	2.3423e-	0	0.9161
		[1.4652]	[1.4789]			52		88		

9	4:30	82.9337	77.1273	5.8064	1	3.3958e-	1	4.7622e-	0	0.2988
		[3.1062]	[2.8244]			24		37		
		32.8854	27.5263	5.3591	1	1.8705e-	1	5.9166e-	0	0.8385
		[1.5756]	[1.6052]			47		73		
10	5:00	86.5115	80.4777	6.0338	1	7.3784e-	1	2.8342e-	0	0.5571
		[4.2234]	[3.8157]			07		08		
		32.1377	26.5481	5.5896	1	9.3867e-	1	6.3783e-	0	0.7029
		[2.0963]	[2.2414]			13		16		

1 reject the hypothesis, whereas 0 cannot reject the hypothesis, Δ mean is the difference between the IMU's and encoder's mean measurement.

The T-test confirms the reading of the two systems does not have the same mean. However, the F-test cannot reject that the two systems have the same variance.

The major reason the analysis matches the order of peak and trough between the two systems is to ensure each system captures the same moment during the trial. One limitation of the actuation of the brace is that it is an open-loop time-based control. It does not control the movement kinematically; therefore, the maximum extension and flexion of the brace could be indeed different every cycle due to the uncontrollable friction in the physical system.

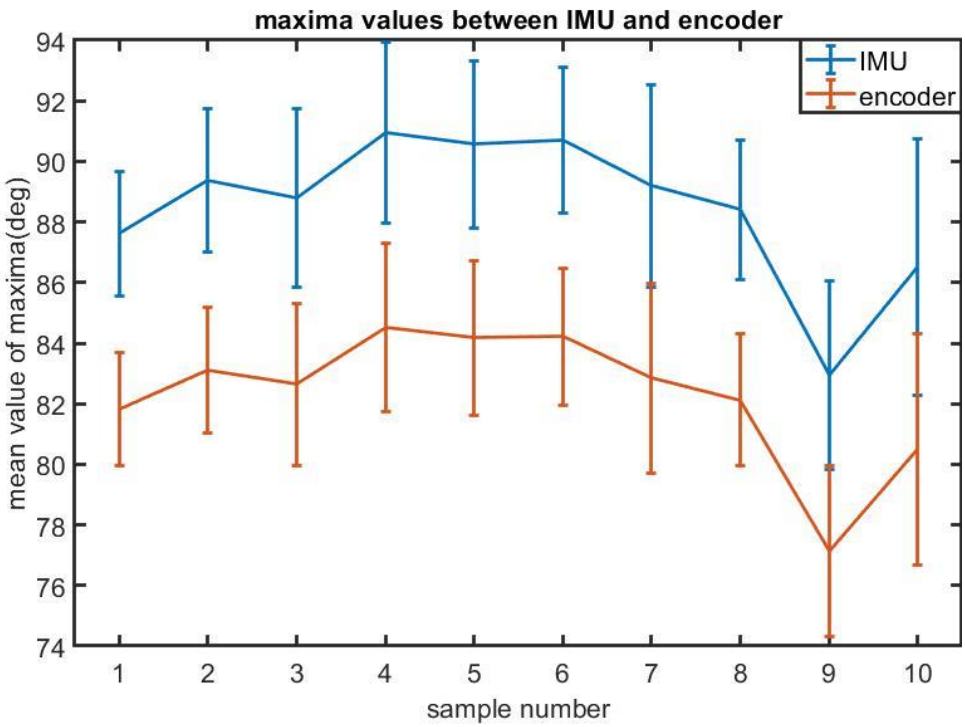


Figure 3.9: The mean and one standard deviation of the maxima of the IMU and encoder measurement across the ten five-minute intervals over the five hours

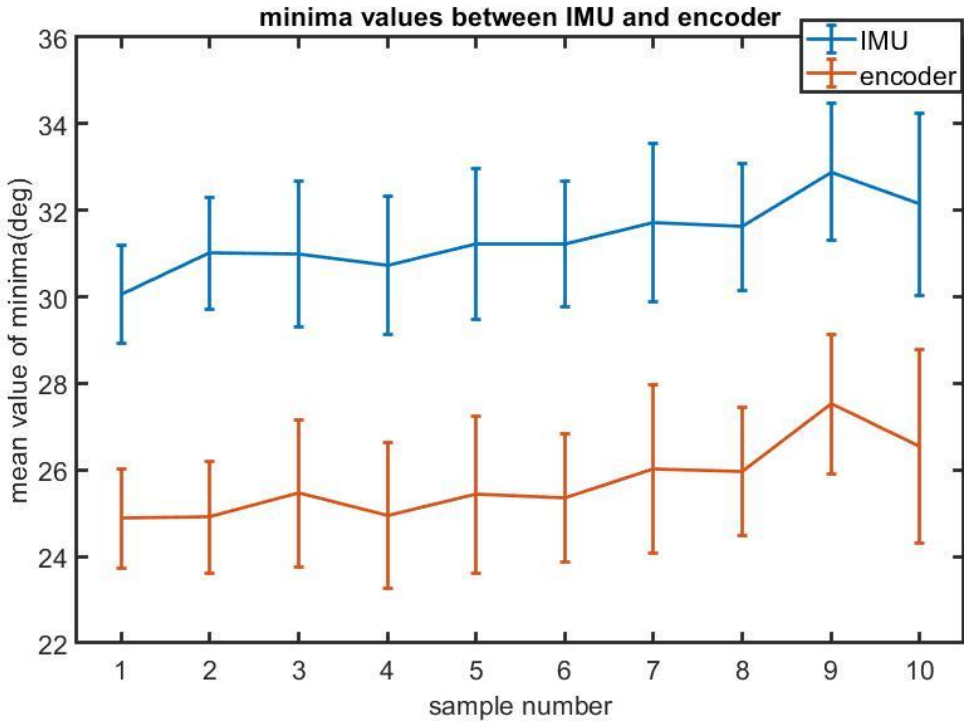


Figure 3.10: The mean and one standard deviation of the minima of the IMU and encoder measurement across the ten five-minute intervals over the five hours

3) Normality test

Normalities are typically tested with normality tests such as Kolmogorov-Smirnov (KS), Anderson-Darling (AD), and Jarque-Bera (JB) test. The basic assumption of these normality tests is that the distribution has enough data to be continuous. The theory of the normality test is based on the expected distribution of values for given sample size or the curvature of the distribution at different regions.

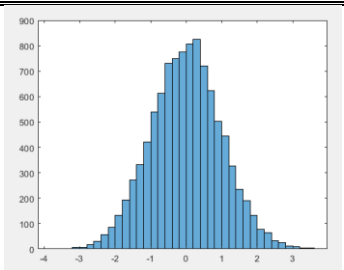
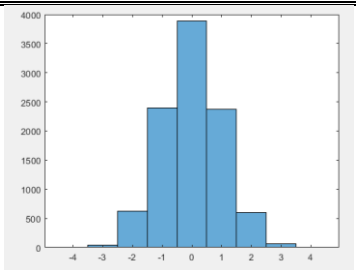
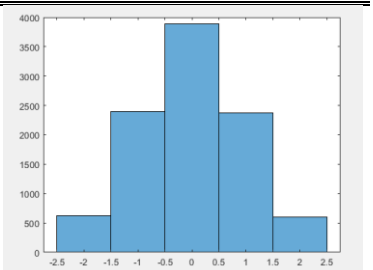
Even if the KS-test reject the noise data belong to a normal distribution in Table 3.4, we could not conclude that there is still a bias in the sensor measurement. This is due to the assumption of normality test that the dataset is continuous, and the measurement error in this study is highly discretised.

The fluctuation of the IMU data is about +/- one to two-bit in value, which make the distribution highly discretised and highly truncated. The method that uses the curvature of the distribution to check normality failed because the discrete values could not reconstruct the bell curve. The methods that evaluate the expected number of samples within each section of the distribution failed for a large dataset because of the absence of data samples with a value above the 2 sigmas range.

The limitation of the normality test is explored with normally distributed random numbers generated in Matlab. An array of normally distributed random numbers are generated using Matlab function, *randn()*, with 10,000 samples. Two arrays are derived from the control array: the first array is its discretisation at each sigma range. The second array has a truncated range excluding any value above two sigmas. These conditions are then tested against three of the most common normality tests available in Matlab, KS test, AD test, and JB test.

Table 3.2.4 shows the p-value of each condition against the normality test listed in the first column. The top row shows the graphical representation of the generated data. Only JB-test can identify the data set is normally distributed after discretisation. All tests failed to identify the sample is normally distributed if the tails are truncated.

Table 3.6: Normality test of randomly distributed random values of 10,000 samples with discretised and truncated condition.

Sample			
	control	Low-resolution	Truncated Low-res
KS-Test	0.5171	0	0
AD-Test	0.4264	5.0000e-04	5.0000e-04
JB-Test	0.2563	0.2804	1.0000e-03

Normality of the current data succeeds when it is broken down into smaller groups around 500 samples because it cannot accurately represent the data in its entirety.

Therefore, there is a research gap in testing the normality of highly discrete digital data. Without the ability to determine if a data set is normally distributed, one cannot be sure that there isn't any skewness in its error.

3.3.3 Performance Outcome

This section has provided a clear evidence for the suitability of using IMU to capture the lower limb movement and interpret the data consistently with gait analysis practices. Three tests are conducted to verify the IMU performance in measuring its orientation data. The quaternion output is transformed into Euler's angle which could be interpreted in the same way as ISB's recommendation for limb movement. These tests have proven IMU to be able to capture limb movement reliably with a proper implementation.

After calibration, the IMU can provide a reliable static reading with reading close to zero and cannot be statistically reject from belonging to a normal distribution with a mean of zero.

The IMU reading is also useful in capturing the relative change in motion during constant movement. A reference system using an encoder on the joint is being compared with the different in roll angle between the thigh and shank IMU. The difference between the IMU roll angles and the joint encoder is no larger than six degrees on average. It is statistically proven that the IMU and encoder measurement belong to the same distribution with a different mean. The ability to capture consistent movement of the limb would allow a consistent feature identification from the captured data to detect gait phases, and hence suitable to the study.

The normality test failed to verify if the fluctuation of the IMU reading is pure white noise. The limitation of existing mathematics is that the dataset must be continuous. The highly discretise fluctuation of the digital signal will make any dataset to be rejected from belonging to a normal distribution with a large sample size.

CHAPTER 4 : ADAPTIVE REAL-TIME DETECTION ALGORITHM

Overview

This chapter introduces a novel adaptive real-time gait phase detection rule-based algorithm based on establishing the normative gait of the user by recording and updating the mean and standard deviation parameters of attached limb kinematics. The algorithms are tested on its intended application using a staircase with 18 steps across 21 healthy participants between 20s and 30s years old. The results show the algorithm performs consistently accurate despite the diversity of height, weight, average cadence among the participants. For consistency sake, the mathematical symbol, ψ would represent the angle measurement in this chapter.

4.1 Algorithm Development

The development of the algorithms consists of two major part. The first is the event selection, which selects a feature associated with the possible gait phases during stair ambulation following the biomechanics description in Chapter 2.2. The second part is the complete architecture and rules definition that guarantee the detection of the selected event in a robust and timely manner. The data used when designing the algorithms are gathered before the experiment and those subject are not invited in the real-time experiment in section 4.2.

4.1.1 Event Selection

Our definition of gait phases in stair ambulation was derived from published biomechanical [104, 108, 110]. Our algorithm uses the outputs from the IMUs on the knee brace described in Chapter 3. It is limited to capture the thigh and shank segment's kinematics only, restricting our definition of the gait phase occurrence to kinematic descriptions only.

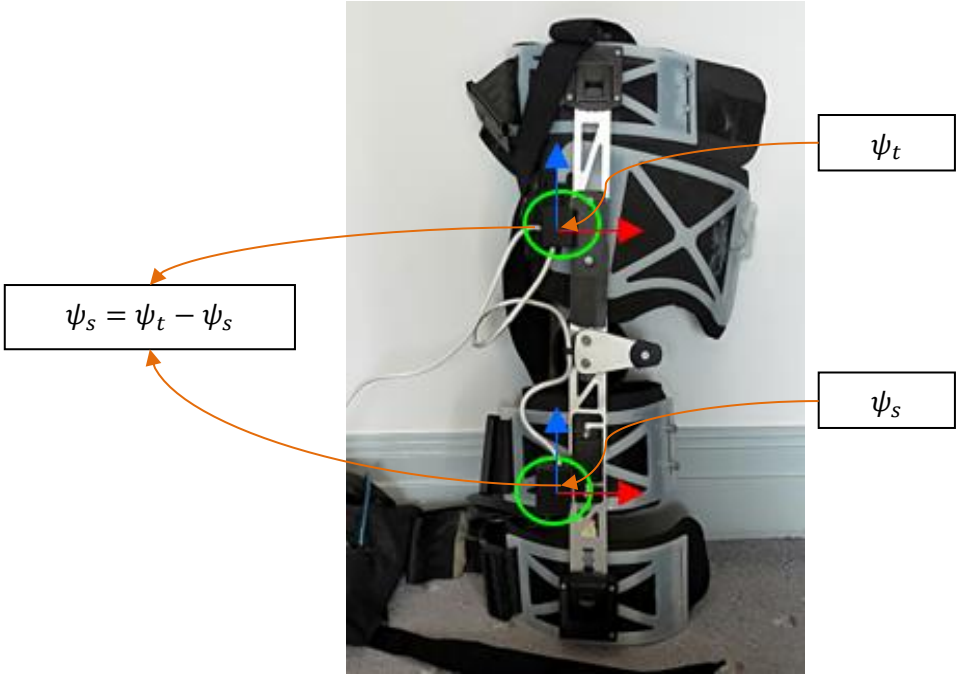


Figure 4.1: Figure shows the angle ψ captured by IMU in the thigh and shank section of the brace. The red arrows indicate the Y axes, the blue arrows indicate the Z axes.

The actual phase occurrence is identified by an assessor from a dataset of five trials of a participant in both stair ascent and descent. Three phases of stair ascent and four phases of stair descent could be distinctly defined by kinematic events from Chapter 2.2 and the sample dataset found them closely matching the identified phases by the assessor. It is assumed that these kinematics events represent the gait phase occurrence as defined in Table 4.1. The selected phases were weight acceptance (WA), foot clearance (FC), foot placement (FP) for stair ascent, and WA, controlled lowering (CL), leg pull through (LP) and FP for stair descent. The psi symbol, ψ, is used to present the angle variable throughout this chapter of the thesis. The subscript of t, s, and Δ represent the object in which the angle is referring to. They are the thigh, shank and their difference respectively, see Figure 4.1.

Table 4.1: Kinematic events for the detection of gait phases

Phase	Stair Ascent	Stair Descent
WA	Min $\dot{\psi}_\Delta$	Min ψ_Δ
CL	-	Local Max ψ_Δ or Inflection point of ψ_Δ
FC	Min ψ_t	-
LP	-	Max $\dot{\psi}_\Delta$
FP	Max ψ_Δ	Max ψ_Δ

WA = Weight Acceptance, FC = Foot Clearance, CL = Controlled Lowering, LP = Leg Pull through, FP = Foot Placement, Min = minimum, Max = Maximum.

4.1.2 Algorithm Design

The algorithms consisted of three stages that adjust the value and timing of each event for each user. The stages were calibration, real-time detection, and continuous step-wise update. The algorithm operates with a sample size of three, refer to (4.1) for the definition. Three is the minimum size required to detect extrema and inflection points, see algorithm formulation in the next subsection. $\psi_i(n)$ is the measurement of i variable, $\{t, s, \Delta\}$, at n program cycle. Then, $t(n)$, is the time stamp at n program cycle. Time derivative was performed after the sample was updated with the newest data. Figure 4.2. provides an overview of the algorithm between the stages.

$$sample = \{(t(n-2), \psi_i(n-2)), (t(n-1), \psi_i(n-1)), (t(n), \psi_i(n))\}, \quad for\ \ n > 2; i \in \{t, s, \Delta\} \tag{4.1}$$

In the calibration stage, the algorithm initiated gait parameters including the range of motion (ROM), $\varphi_{rom,i}$, the values and timing for the maximums, $\varphi_{max,i}$ and $t_{max,i}$, and minimums, $\varphi_{min,i}$ and $t_{min,i}$, of ψ_t, ψ_s, and ψ_Δ and their derivatives of the first complete step. $\varphi_{j,i}$ and $t_{j,i}$ are respectively the value and timing of j feature of i variable. ROM was defined by the difference between the maximum and minimum measurement angles during the first complete step.

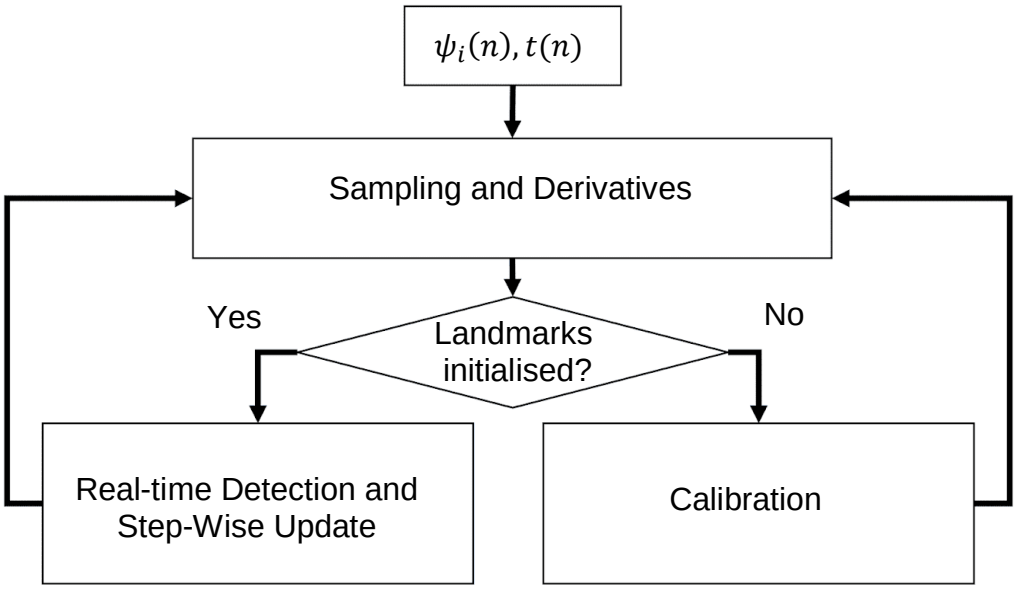


Figure 4.2: Top-level flowchart diagram of the algorithms: from the sampling and derivatives of input data to the calibration stage, and real-time detection and step-wise update. The input data are the current timestamp, $t(n)$, and measurement, $\psi_i(n)$, where n is the program cycle, and i is type of variable

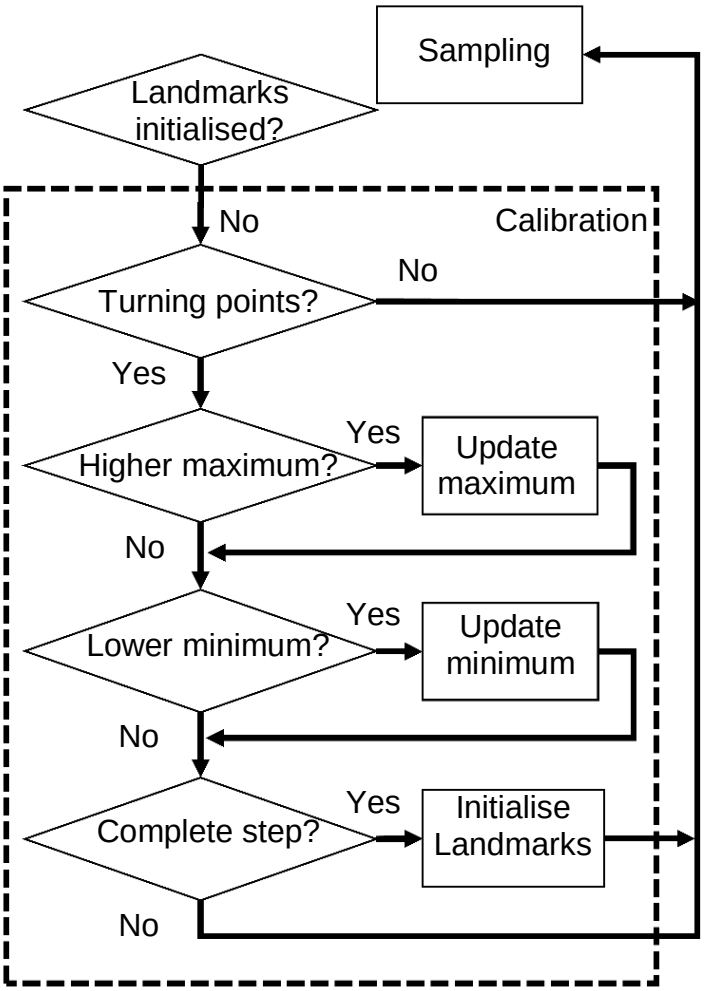


Figure 4.3: The calibration stage: Maximum and minimum of the ψ_t, ψ_s, and ψ_Δ and their derivatives are continuously monitored until one complete step is detected after all other turning points are found in other variables.

Since the algorithm had no previous knowledge of any gait parameter, the first complete step was determined by a sequence of events. The algorithm checked for the first occurrence of a ψ_Δ maximum; this marked the beginning of the calibration cycle. Then, it populated the critical values of all other events. All maximum and minimum values must be found after the occurrence of a ψ_Δ maximum, for the second θ_Δ maximum to mark the end of the first step and initialise the first gait cycle time, T (5), and the ROM of all variables. $T(N)$ is the gait cycle time of the N gait cycle. If the next ψ_Δ maximum was found before other events. It indicated the previous ψ_Δ maximum is not the true maximum and resets the calibration cycle. Figure 4.3. shows the flowchart diagram of the calibration stage.

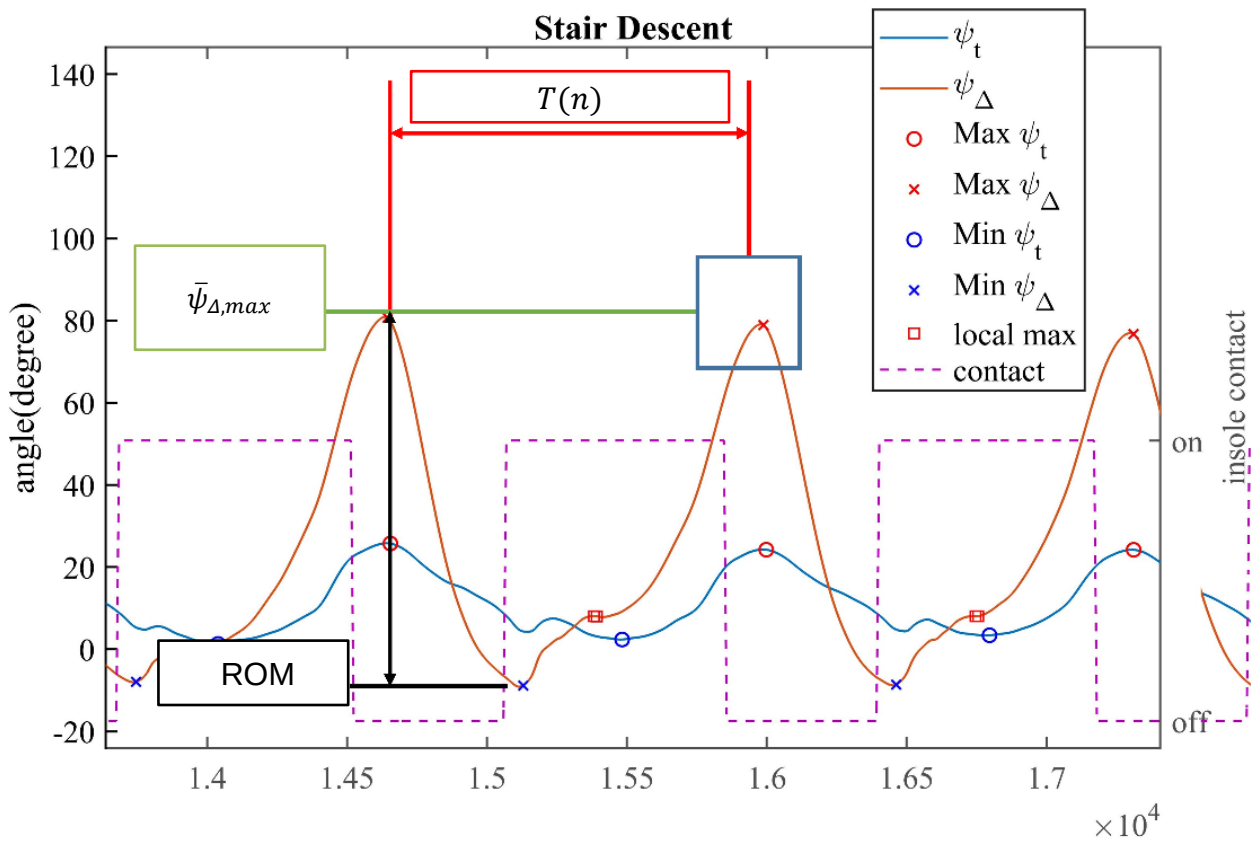


Figure 4.4: A real-time detection example: the algorithm uses previously established parameters to search for the first instance of event occurrence within the threshold window.

$$T(N-1) = t_{max,\psi_\Delta}(N) - t_{max,\psi_\Delta}(N-1), for\ N > 1 \tag{4.2}$$

During the real-time detection stage, the algorithm calculated the thresholds for each targeted event based on the parameters established in the calibration and searches for the specific event within its threshold window. The threshold for the kinematics value of the event was set at ±10% of the maximum ROM of each respective variable. The threshold for the timing of the event was ±10% of the most recent gait cycle time from the previous instance. Figure 4.4. shows an example of FP detection for stair descent. The threshold window (the blue rectangular box) is the region of interest of the next FP phase occurrence based on existing parameters. The green line represents the current mean of maximum ψ_Δ, hence the threshold angular boundary of the threshold window is for the next peak detection is ±10% of the maximum ROM around that mean values. The ROM is shown with the black double arrow line. The red double arrow line represents the projection of the most recent gait cycle time, $T(n)$. Then, the timing boundary of the threshold window for the next peak detection is ±10% of that gait cycle time from the current peak detection instance. The first event detected within this window will be the beginning of the next FP phase. This procedure of thresholding and estimation of next detection is done for each kinematics features of each

variable.

In the event of a missed detection, the algorithms took an offset from the next detected phase to estimate the occurrence of the missed phase in accordance with the gait partitioning found in the literature [104]. For example, a missed FP had occurred in stair ascent because the next possible IC was detected before the maximum of ψ_{Δ}, and the timing was outside the 10% tolerance. The missed FP was then assumed to have occurred at 18% of the gait cycle time before the newly detected IC point. The gait cycle time would not be updated, because the assumed occurrence is not a detection of the maximum of ψ_{Δ}.

For stair descent, the CL phase occurred near an event that is not established in the calibration stage, and therefore it is treated as a missed detection during the calibration stage. The procedure for a missed event described in the previous paragraph is applied during the real-time detection stage.

A step-wise update operated in parallel with real-time detection. This update aims to establish the normative gait parameters for the user. Upon each successful detection, the algorithm updated the respective parameters by taking the mean in both the measurement values and their timings using (4.3), where P_j is the parameter of the j event, and P is either the angle measurement or the timing; n_j is the number of occurrences of j event. The ROM of the new cycle will be compared to the existing ROM, and updated if the new ROM had increased. This is to account for the transient step from standing, which has a smaller peak value than a progressive step, during the calibration stage. The gait cycle time was updated to the time difference between two successful consecutive ψ_{Δ} maximum detection (FP phase). Figure 4.4. shows the flowchart diagram of the real-time detection and step-wise update stage.

$$\overline{P_j} = \frac{(n_j \overline{P}_j + P_j)}{n_j + 1}, \text{ Then } n_j = n_j + 1 \tag{4.3}$$

Long-term memory storage was available to store the parameters calculated in previous trials. The algorithm could use previously saved parameters instead of re-calibrating the first step of the next trial. This feature would allow the algorithm to respond to the first step of the stair gait if needed. Since we would like to verify the function of the calibration stage, we

decide not to use this feature for this study.

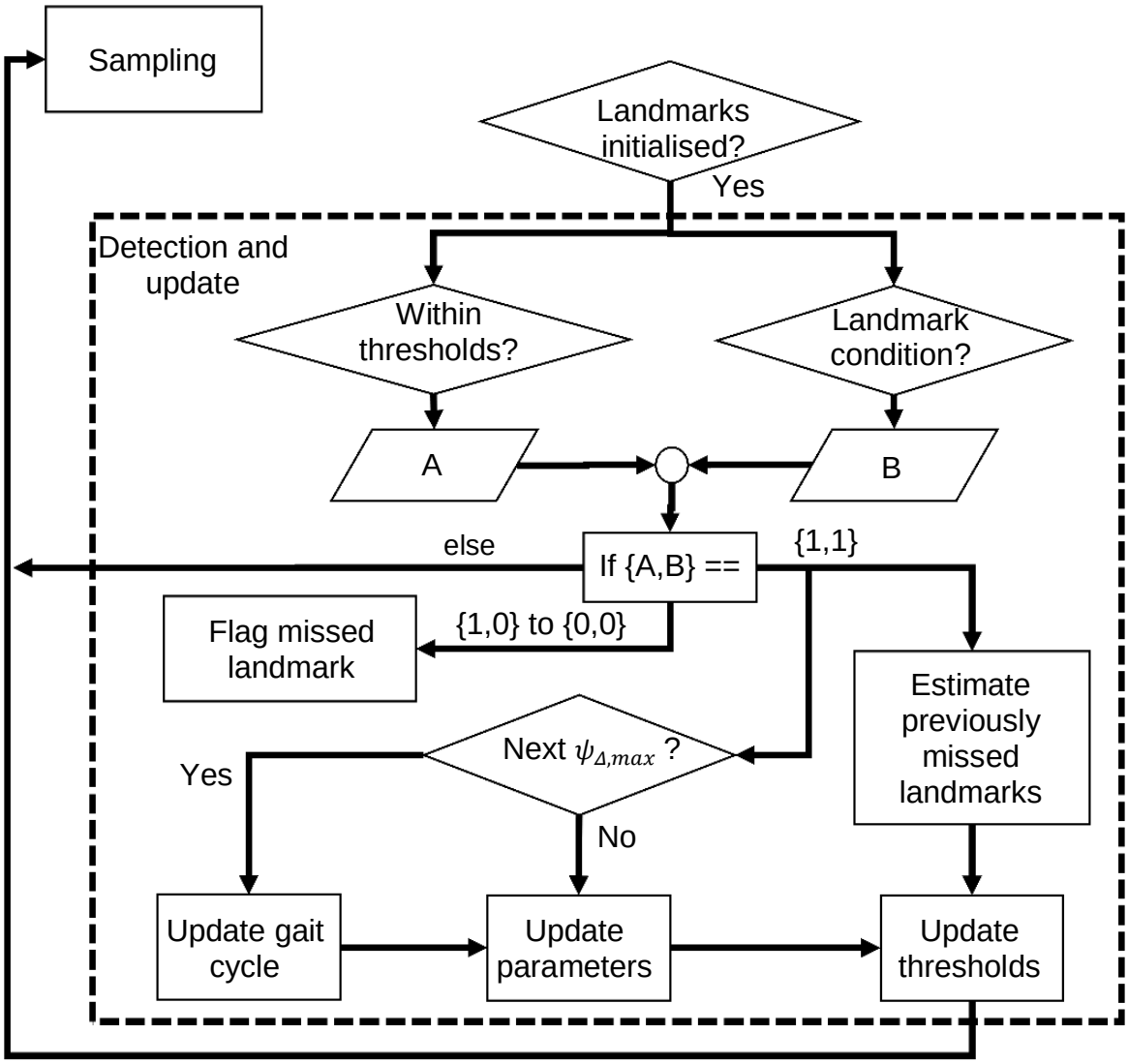


Figure 4.5: The real-time detection and update stage: the sampling data are checked with both the event feature and the threshold windows. A missed event is flagged if no event is detected within the entirety of the threshold. A successful detection in both will update all parameters and thresholds accordingly. Gait cycle time update when next ψ_{Δ} maximum is detected.

4.1.3 Algorithm Formulation and Implementation

The previous subsection has layout the kinematics event and the general workflow of the algorithms. The specific condition for detecting a phase can be generalised as a condition of the turning points and the inflection points within an adaptive window. The conditions can be formulised and modularised, and then applied to each targeted kinematics event.

It is possible to determine turning point and inflection point by examining the first and second derivatives of the dataset. Hence, three data sample is the minimum number to determine

the nature of these critical points. The mathematical definition of maximum is where the derivative function changes from positive to negative. Minimum is where the derivative function changes from negative to positive. Rising inflection point is where the second derivation function changes positive to negative. Falling inflection point is where the second derivation function changes negative to positive.

Since our data is digitalised, and we cannot guarantee the turning points to be stationary, we will have to observe the rate of changes on both side of the data samples. The direction of the rate of change is observable by comparing the consecutive data sample. Without the exact calculation of the derivative increases the algorithmic efficiency of the logical operation. There may also be the possibility to have a saturated turning point due to the digitalised data. We considered a saturated turning points by allow an equal value between the current sample and the previous one. This saturation condition does not apply to inflection points, so a saturated inflection would be recognised as a local saturated extremum followed by a flat line. Given the variable we are observing for inflection points is the first derivative of the angle data, the detection of this point is inherited one sample slower than the turning points. The definition of each type of critical points are summarised in Table 4.2 below.

Table 4.2: Conditions of critical points

Type	Condition
Maximum	$\psi_i(n-2) < \psi_i(n-1)\ \&\ \psi_i(n-1) \geq \psi_i(n)$
Minimum	$\psi_i(n-2) > \psi_i(n-1)\ \&\ \psi_i(n-1) \leq \psi_i(n)$
Rising inflection	$\dot{\psi}_i(n-2) > \dot{\psi}_i(n-1)\ \&\ \dot{\psi}_i(n-1) < \dot{\psi}_i(n)$
Falling inflection	$\dot{\psi}_i(n-2) < \dot{\psi}_i(n-1)\ \&\ \dot{\psi}_i(n-1) > \dot{\psi}_i(n)$

A flag is available for the detection of each type of critical points. The algorithm is constantly checking and updating these flags during operation. The flag of these critical points is the B signal output of the landmark condition in Figure 4.5 for their respective gait phase in Table 4.1.

The output of the threshold window is parametric condition of the current sample against the stored historical value of those tempo-spatial parameters. The historical vale for the parameters is created from the calibration stage. The logical condition of finding the maximum and minimum landmarks within the calibration step is defined in Table 4.3. $P_{j,max}$ is the most recent occurrence of a maximum of P_j parameter, where as $P_{j,MAX}$ is the

historical value of the maximum of P_j parameter during the gait cycle. The calibration stage records the maximum and minimum of each variable in one gait cycle. It is an important process that make the algorithm adaptive to each user. It set up all the parameters required to the following detection in the real-time detection stage.

Table 4.3: Conditions of updating turning points in calibration

Type	Condition
Maximum update	$P_{j,max} \geq P_{j,MAX}$
Minimum update	$P_{j,min} \leq P_{j,MIN}$

The implementation of recording the maximum and minimum of each variable is done by building a class structure that has a list of property including the turning points value, the mean of those turning points, the most recent timing of the turning points. The class structure can be passed to the gait phase with the appropriate kinematics events. This way we can use the same process for different landmarks for different phases. This data management would allow a smaller memory as information are recycled and shared, while protecting the relevant data of each gait phase from other gait phases that use the same critical point such as FP for stair ascent and descent. Both FP for ascent and descent are using the same variable for detection, so they would have different record of the timing and value for the detection.

```
Class critical_point {
Public:
      long int occur_time;
      float recent_value;
      float mean_value;
      int occurance_count;
      void update(void){
            mean_value = (mean_value* occurance_count + recent_value)/
      (occurance_count+1);
            occurance_count = occurance_count+1;
      }
};
```

The modularised implementation of each detection condition and for the specific landmark for a gait phase, allow real-time operation of the detection algorithm on a microcontroller with limited computational power and memory.

4.2 Experiment

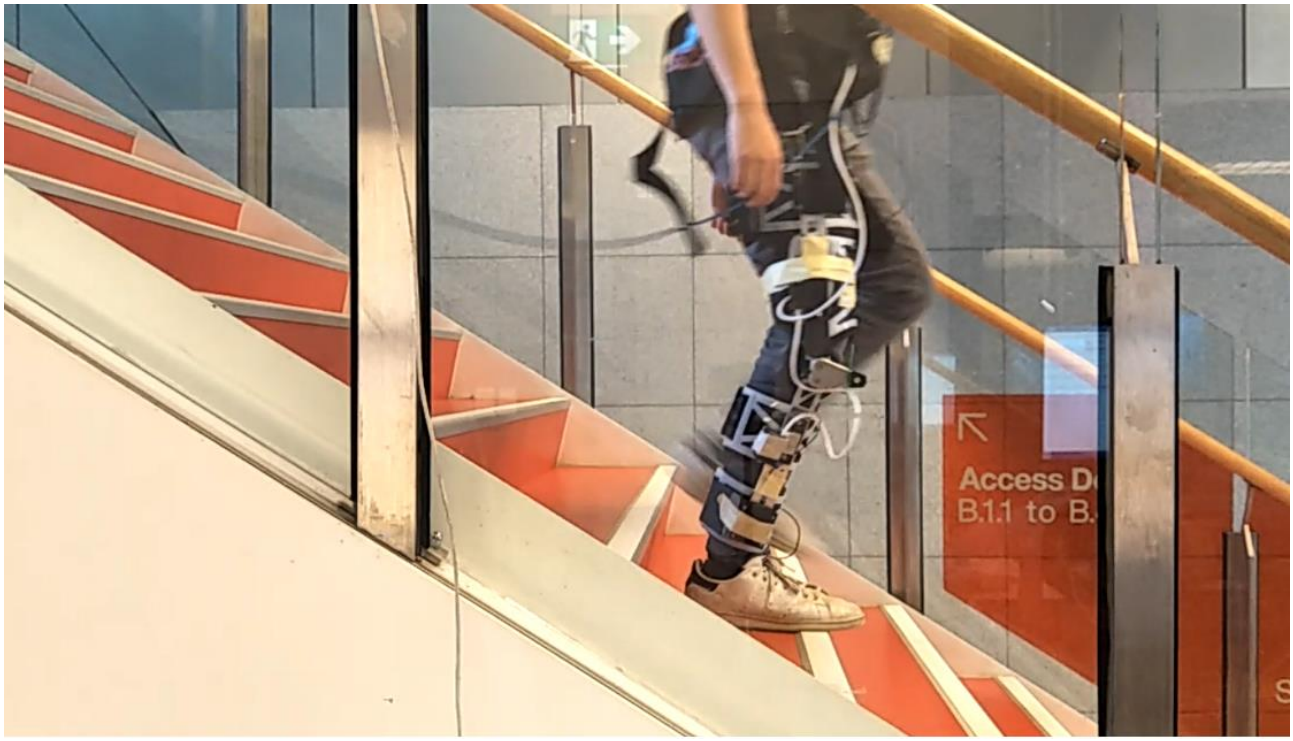


Figure 4.6: A participant walking down a staircase in an out-of-lab environment wearing the measurement brace

A convenient sample of 21 healthy participants was recruited (18 males) from the university community. Participant profiles are summarised in Table 4.4. Participants were excluded if they had previously been diagnosed with any neurological or orthopaedic condition, had a history of lower limb joint surgery, were currently experiencing lower limb pain, or had recently suffered a lower limb injury. A minimum sample size of 20 (significance level of p=0.05 and power=0.8) was required to establish a control group according to previous biomechanical gait studies reported in the literature [165, 166].

Demographic data were collected, including age, height, weight and leg length. Leg length was taken as the average value across three separate measurements using a tape measure from the anterior superior iliac spine (ASIS) to the medial malleolus [167]. No participant was found to have a leg length discrepancy.

Table 4.4: Participants Profiles

Subject	21
Age	26.14 (3.53) [22,34]
Height (cm)	171.93 (8.61) [152.5,189.5]
Weight (kg)	64.99 (10.31) [50.5,90.5]
Left Leg Length (cm)	90.09 (5.81) [78,98.5]

Right Leg Length (cm)	90.05 (5.83) [78,98.5]
Average Cadence of Stair ascent (steps/s)	0.85 (0.09) [0.67,0.99]
Average Cadence of Stair descent (steps/s)	0.91 (0.11) [0.68,1.13]
Lefts Leg Length/rise*	5.30 (0.34) [4.59,5.79]
Right Leg Length/rise*	5.30 (0.34) [4.59,5.79]

Numbers in the cells are represented as the mean (standard deviation) [range] of the variables. *Length over rise is a dimensionless ratio between the participant leg length and the rise of each step of the staircase.

The measurement brace was attached to the right leg of each participant. The insole footswitches were attached to the sole of each foot using double-sided tape. The participants then donned their shoes. The response from the insole footswitches was then tested. Fitting of the sensors/shoes was adjusted if the signal was delayed or deemed too noisy. The waist pack carrying the onboard microcontroller was then fitted to the participant. The data collected by the microcontroller from the IMUs, algorithm outputs, and insole footswitches signals were transmitted through a USB cable to a PC. Each trial was recorded on video for post hoc observation if required. No quantitative result was calculated from the video data.

The participants performed the stair trials on an 18-step staircase, with each step rise 17 cm and run 27 cm. The staircase had a handrail on each side. The participants were advised to use the handrail if they felt they were at risk of falling. Participants were instructed to perform a step-over-step gait at their preferred speed. This study focuses on normal stair climbing gait; hence the transient steps, stumbling, or other non-stair climbing steps were removed from the data analysis.

Each participant was given five to ten minutes to walk with the brace before practising stair ascent and/or descent. Once the participant was familiar with the device and the procedures of the experiment, testing commenced. Participants were given the opportunity to rest at any time during the experimental protocol.

Prior to the first trial, the participants were asked to stand still in an upright position before powering up the measurement system. The trial commenced with a push-button on the command of the researcher. Ascent and descent trials are conducted separately.

4.3 Data Analysis

The detection of the phases from the algorithm was operated in real-time, whereas the ground truth from the insole footswitches was determined in post hoc analysis. An 11-order moving median filter was applied to the insole signals. False-positive contacts of the footswitches during the swing phase and false-negative contacts during the stance phase were manually corrected. We considered IC is the activation of any switch inside the insole, whereas EC is the deactivation of all switches.

The performance of the IMU-based GPD algorithm was evaluated with recall, precision, F1-score, and the timing error against the reference insole footswitches signals. Recall or the true positive rate (TPR) defined the true positive detection among all true positive and false negative detection signals for a specific gait phase. Precision or the positive prediction value (PPV) defined the true positive detection among all positive detection for a specific gait phase. F1-score was the harmonic mean of TPR and PPV. The timing error was calculated by subtracting the time instant of ICs or ECs recorded by the insole switches from the time instant that the algorithm detected this event. We analysed the timing error in both exact and absolute value. Therefore, a negative value indicated an early detection, whereas a positive value represented a delay. We also analysed the variation (the standard deviation) of the timing error to show how consistent the detection timing was for each subject.

4.4 Results

All 21 participants completed the testing. A total of 524 trials (251 ascent and 273 descent) were collected, with 3419 steps (1665 ascent) included. Each step was defined between two IC instances of the insole footswitches; therefore, 3943 IC instances were included in the data analysis.

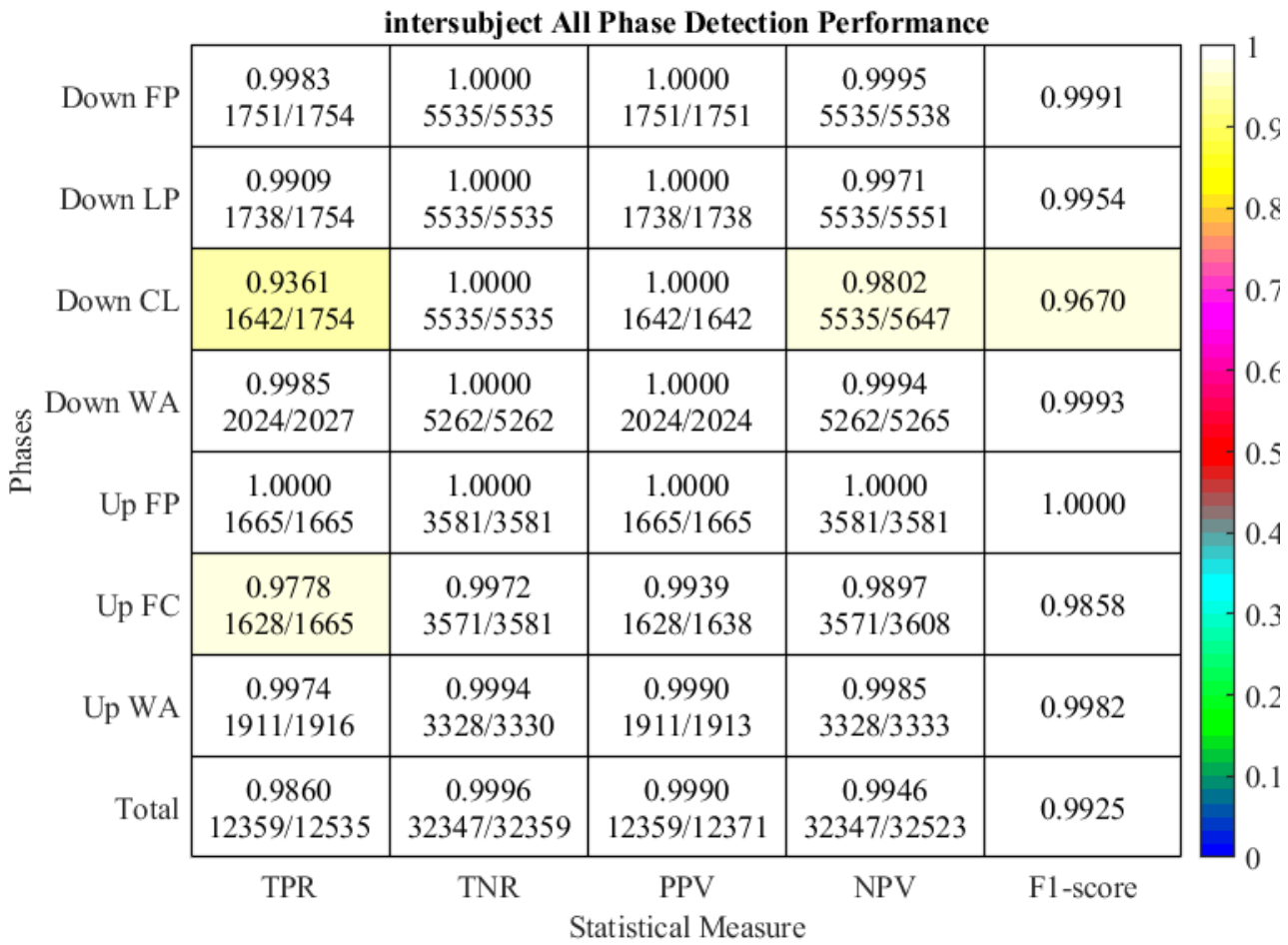


Figure 4.7: The detection performance of the algorithm for each selected phase. TPR = true positive rate (recall), TNR = True negative rate, PPV = Positive prediction value, NPV = negative prediction value

Figure 4.7 provides the successfulness of the detection of each phase. This was calculated based on the number of detections by the algorithm of each phase, the number of true positive detections by the algorithm, and the number of actual occurrences of each phase. The CL of stair descent had the lowest F1-score and the lowest TPR. However, CL had a PPV of 1, meaning all the errors were false-negative predictions. The lowest PPV (0.9939) occurred in FC of stair ascent. The best performing detections occurred in the FP and WA phases for both stair ascent and descent, with ascent FP reaching an F1-score of 1 on 1665 occasions.

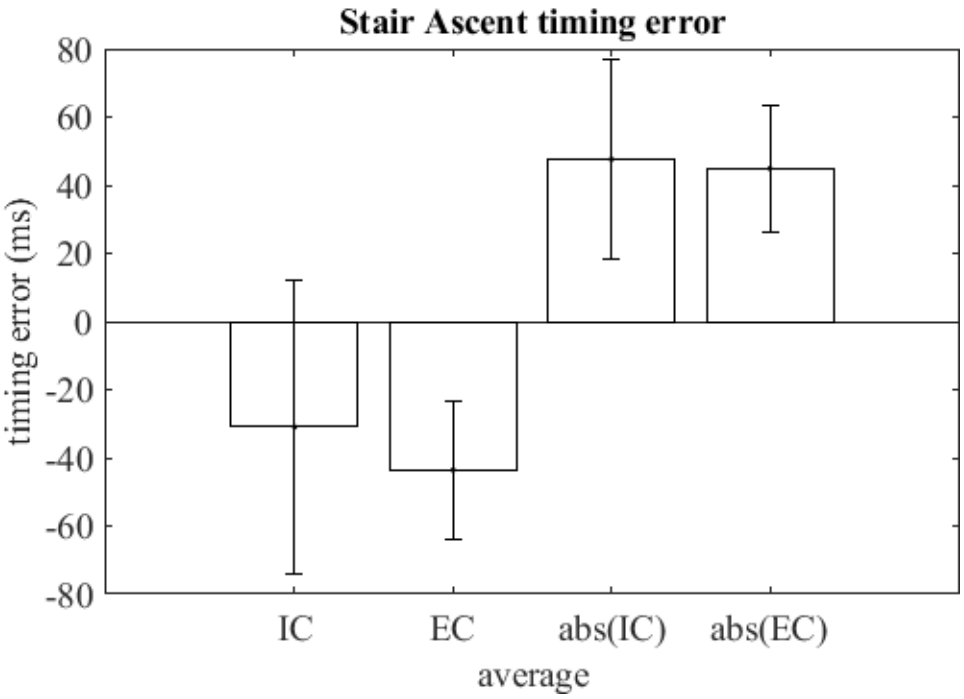


Figure 4.8: The average timing performance of the algorithm for each reference event (IC or FO) in stair ascent across all subjects. Error bar of each bar represents the standard deviation.

For stair ascent, the inter-subject means of the participants' mean errors for IC and EC were -30.7 and -43.66 ms, respectively. The inter-subject means of the participants' standard deviations for IC and EC were 33.86 and 18.22 ms, respectively. Most detections occurred early for EC with only one participant showing a more evenly distributed detection around the actual occurrence. Figure 4.8 provides the pooled mean timing error in stair ascent and the standard deviation across all subjects.

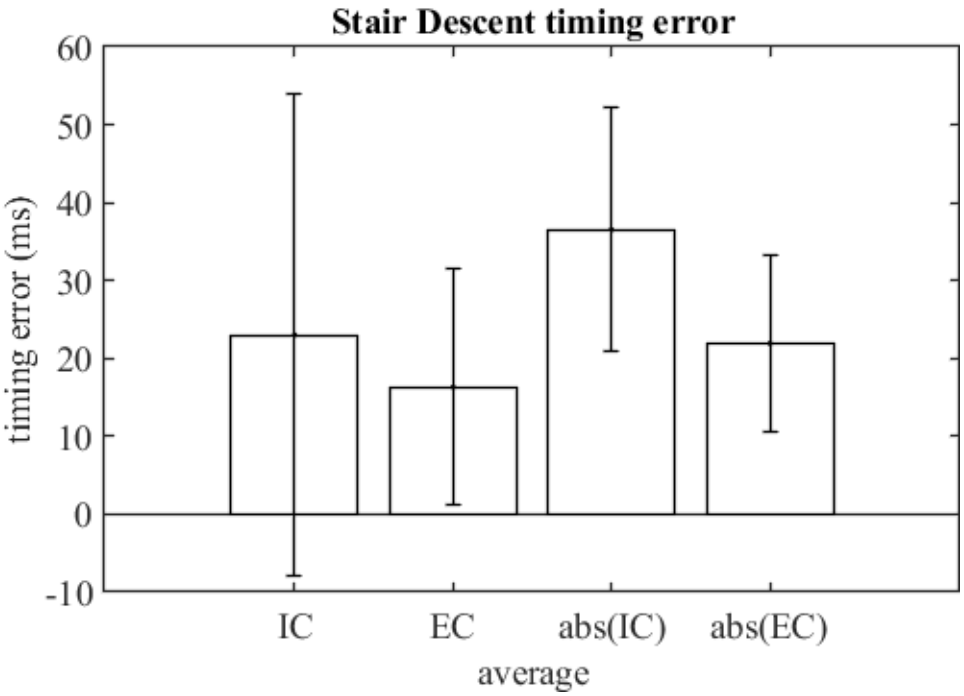


Figure 4.9: The average timing performance of the algorithm for each reference event (IC or EC) in stair descent across all subjects. Error bar of each bar represents the standard deviation.

For stair descent, the inter-subject means of the participants' mean errors for IC and EC were -23.43 and -16.41 ms, respectively. The inter-subject means of the participants' standard deviations for IC and EC were 19.85 and 14.86 ms, respectively. The detection occurred mostly after the actual event. Only five participants had a significant portion of early detection for IC. Figure 4.9 provides the pooled mean timing error in stair descent and the standard deviation across all subject. Table 4.5 summarises the inter-subject mean and standard deviation of the participant means and standard deviations of error.

Table 4.5: Detection Timing Errors and Variations

Variable	Inter-subject mean timing error in ms		Inter-subject mean timing variation in ms	
	Ascent	Descent	Ascent	Descent
IC	-30.70 [43.25]	23.43 [30.32]	33.86 [10.35]	19.85 [8.88]
EC	-43.66 [20.12]	16.41 [15.23]	18.22 [7.70]	14.86 [5.47]
Abs(IC)	47.75 [29.31]	36.60 [15.56]	27.97 [9.55]	16.66 [6.83]
Abs(EC)	45.00 [18.65]	21.87 [11.29]	17.04 [5.88]	11.39 [3.03]

Numbers in the cells are represented as the mean [standard deviation] of the variables

4.5 Discussion

The study aimed to develop and verify the algorithm ability to correctly detect targeted gait phases within the requirement of ±50 ms from the actual occurrence. Our results have a mean timing error below 50 ms and the standard deviation below 20 ms except for IC of stair ascent. Assuming our timing error belongs to a normal distribution with the mean and sigma of the inter-subject mean and mean standard deviation listed in Table 4.4, the likelihood of a detection outside the 50 ms range from the actual event are 29.29%, 9.05%, 36.39%, and 1.19% for ascent IC, descent IC, ascent EC, and descent EC, respectively. As early detection can be artificially delayed, the mean could be offset to shift the distribution around zero. In this scenario, the likelihoods for a detection outside the acceptable range are 13.98%, 0.61% for ascent IC and EC. The adaptive approach taken in this study has allowed the algorithm to operate with consistent accuracy despite the variation in gait speed among the participants, from 0.69 to 0.99 steps/s for stair ascent, and from 0.68 to 1.13 steps/s for stair descent.

It is expected to have an overall delayed detection, since all sensor are measurement taken after the occurrence of the event. The filtering and data fusion of the IMU have limited the

output of the measurement to 100Hz, which there is at least 10 ms of measurement delay. This inherited delay is further combined with the motion of the measurement brace is driven by the leg it is attached to. The soft tissue of the leg and the compliance of the brace will act as a spring in series and damping and delayed any change of motion physically. Future device could consider a convenient method of attaching the sensor on the user's limb to better reflection the true motion of the user.

Most detection errors in our algorithm occurred from false negative, where the events were outside their threshold windows. This may be related to a possible change in the speed of movement or range of motion. The consistent early detection of stair ascent IC may indicate that our selected event may not be closely associated with the actual IC event. Formento *et al.* [96] found a close relationship between the $\dot{\psi}$ in the local IMU sagittal plane to the IC instance.

Previous GPD studies have reported a larger variation in the detection of EC instances [28, 29]. The results of this study, however, have a larger variation in the detection of ICs, especially for stair ascent. This inconsistency may also suggest that the occurrence of minimum $\dot{\psi}_{\Delta}$varies widely across participants. The inconsistency may also suggest that there may be different modes of contact for stair ascent gait. From the observations of the data, the actual IC could occur between the minimum of $\dot{\psi}_{\Delta}$ and the minimum of $\dot{\psi}_{s}$, as shown in Figure 4.10 and 4.11. Further gait analysis in alternative gaits of stair ambulation is required to understand the underlying reason for the difference observed, which is beyond the scope of the present study.

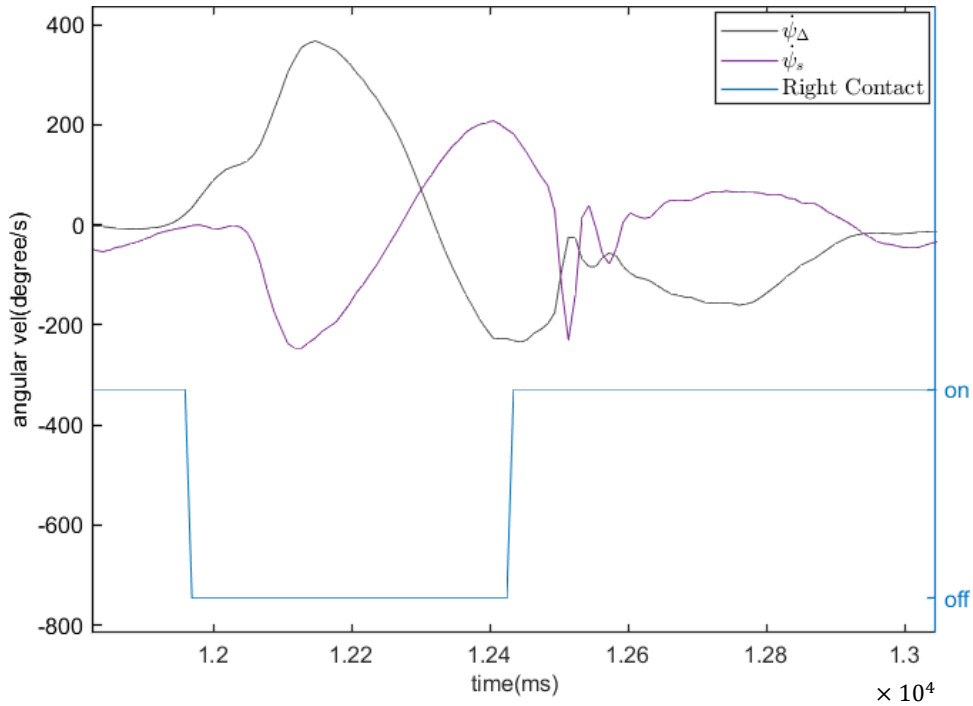


Figure 4.10: An example of IC instance of subject 3, where the IC instance is close to the minimum of $\dot{\psi}_\Delta$.

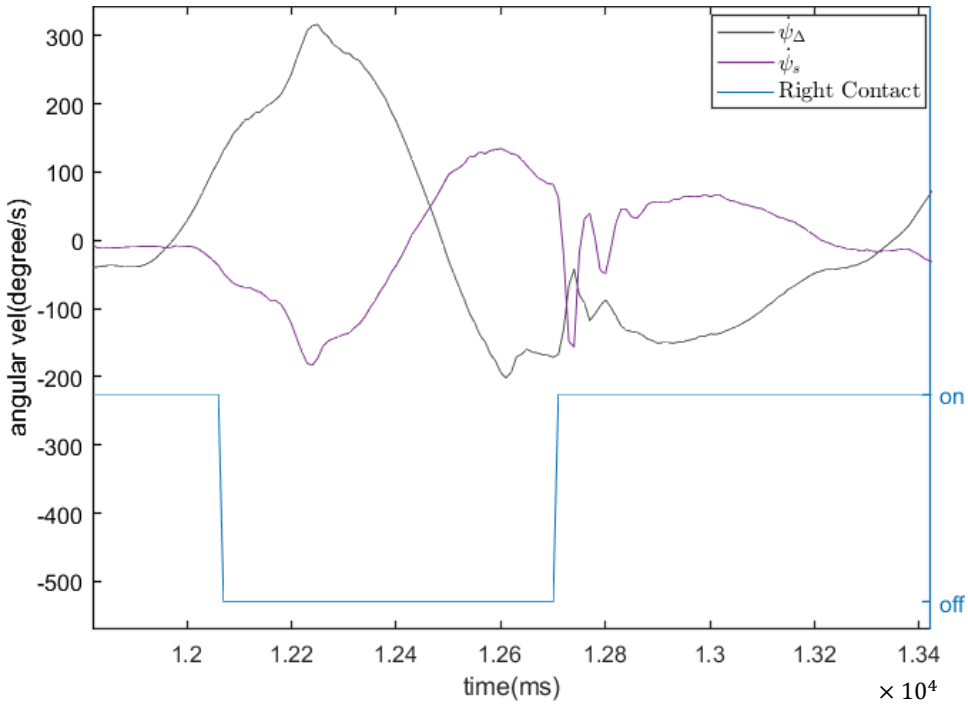


Figure 4.11: An example of IC instance of subject 6, where the IC stance is close to the minimum of $\dot{\psi}_s$.

An overall F1-score of 0.9925 across all phases in both stair ascent and descent indicates that the thresholding has a high probability of locating the range where the next event can occur. It illustrates knowledge of gait biomechanics is crucial in designing a detection algorithm for gait phases. The definition of gait phases in the literature heavily uses

biomechanical variables such as thigh, knee and shank movement. In this study, we interpret the IMU rotations on the segment of the knee brace to estimate these biomechanics variables. Future research could focus on analysing accurate body movement from attached IMUs similar to the OpenSense project in OpenSim4.1 leading by Prof Scott Delp but for real-time application.

The chosen test environment is a flight of staircase with multiple progressive steps, where a user of an assistive device may need support the most. The evaluation of the detection performance in this environment has direct relevance to its actual application. Despite the close relevance the results have on its application, the test was performed outside a lab environment. Therefore, the typical gold standard of using a force plate and 3D motion capturing is impractical. Insole footswitches were used as a substitute for force plate as the ground truth for IC and EC [55, 68, 99], but a substitute for a 3D motion capture system was not available. Hence, the timing errors from the ground truth for our algorithm only includes IC and FO instances; the exact timing error of the transition to CL and FP phases are unknown. An established measurement system that accurately captures body movement outside a laboratory environment would allow a more comprehensive study to be conducted in a realistic environment. Validation of using IMUs as the alternate kinematics measurement for analysis purpose remains a research gap in the field.

We avoid using dimensionless results such as error in percentage cycle as suggested by McGinley *et al.* [48]. Measuring the error per gait cycle will heavily favour participants who have a slow gait. Furthermore, the detection algorithm looks for kinematic events in the data, and the detection timing error should be independent of how fast or how slow the gait cycle is. A requirement of the responsiveness of assistive devices that are expressed in percentage gait cycle would have limited usefulness apart from detecting the range of gait speed for which a specific device is suitable.

This study extends previous GPD work by exploring the use of biomechanics variables such as the thigh, knee and shank angle in the sagittal plane, and attempt to bridge the knowledge between the two fields. Many previous gait phases/event detection studies examined variables that are directly related to the raw data of the IMUs. The method of using the direct variables from the sensor reduces the number of mathematical computations for any intermediate variables. However, the choice of not using joint and limb movement limits the ability to translate existing knowledge in biomechanics into the rules of detection, as many

records of gait patterns are described in terms of joint variables. The selected event for each phase can easily change to a different feature of another variable. Further finetuning of the algorithm may include feature engineering for a more consistent variable for the rules.

The feature-based detection approach of this work can be applied to detect other events in the gait cycle or other activities. Hence, it can be readily integrated with other existing rules found in literature or applied to a different joint with different events or for different activities. Future development may extend the application of the current detection approach to another joint such as the ankle and hip or other activities such as level walking and ramp walking.

It is important to point out that the developed algorithm is for stair climbing gaits only. This algorithm could be incorporated into an activity detection such as Lau *et al.* [76] and Archer *et al.* [102] so that the program can determine which gait activity the user is engaging and then select the appropriate gait phase detection model for assistive control. It is reported that most commercial devices require the user/therapist to select the activity [12]. The automatic detection and classification of gait activities in real-time on wearable devices remain a research gap.

4.6 Conclusion

This work developed algorithms to detect stair gait phases in real-time using IMUs. The algorithms deploy a 3-stage process to establish the normative gait and adapt to different users during the activity without prior knowledge of the user. The algorithm was implemented and tested on participants in real-time, showing promising results based on the overall F1 score of 0.9925 of using IMU data to detect some of the stair gait phases.

We showed that translating the existing biomechanics knowledge for gait phase detection is crucial as part of the design process. This approach resulted in a highly repeatable detection of gait phases using kinematics events measured from IMUs. The mean standard deviation for detecting IC and EC is under 20 ms except for stair ascent IC across participants with a wide range of cadence, from slowest of 0.67 to faster 1.13 steps/s. The large variation in stair ascent may be due to the different strategies of foot contact by the participants.

The results of this study showed that our algorithms are feasible and can be used in a wearable assistive device. The algorithms were implemented and tested on a knee device in a realistic environment; a staircase in a building with multiple progressive steps. The developed algorithms and the sensory system are readily implementable onto most commercially available knee braces.

Several research and technical gaps have been identified and discussed that require further investigation beyond the scope of the current study. These include the consideration of alternative stair ambulation, a measurement system that accurately captures body movement in an outdoor environment, the implementation of the current detection approach for other joints, and the integration with gait activity detection or the integration with falling or tripping detection.

CHAPTER 5 : MACHINE LEARNING APPROACHES

Overview

This chapter introduces a comparison of the different machine learning techniques. This section of the study explores the performance of multiple common machine-learning techniques found readily on MATLAB 2020a/b. It would enable developers to make a better-informed decision when choosing the technique that is best suited to them. Supervised learning classifier and two suitable architecture of recurrent neural network are selected in this study. Since gait phase detection is the detection of state transition in the output, we also investigate the effect of having a labelled output on the state transition instead of the actual state of stance and swing phase. Some trained models have displayed a high level of correct detection and small timing error for different activities and either stance or swing.

5.1 Data Preparation

The data gathered from the rule-based detection study is reused in this comparison study to evaluate each technique's performance against the rule-based approach. We decided to divide the dataset in half for the training and testing data. We prepared the data of each trial, so they are trimmed to begin and end on initial contacts. This would allow the training to learn from completed labelled steps only and avoid incomplete cycles.

Then, we categorise the data into each subject in each activity, a total of 42 sets (one for each subject in each activity). A randomly selected subject-based dataset was included in the training data for each specific activity until half of all steps for each activity are included. We used the same set of training data and testing data for both the neural network and supervised learning models to compare their performance on the same data.

There are three types of output we are going to train for: 1) the state output of the footswitches that indicate the stance (1) and swing (0) phase, this is the control test, 2) the transition output of the initial contact (1) and foot off (-1), other instances are steady-state(0), 3) 5-sample-wide transition output of the initial contact (1) and foot off (-1), other instances are steady-state (0). Figure 5.1 gives an example of the output signal of each output type in one gait cycle. The purpose of training for transition output is because this is the output we

ultimately seek when evaluating the timing difference of initial contact and end contact of each step. Pinpointing the output to the exact moment of transition, we expected to reduce the timing error of a positive prediction when the trained model tries to reproduce the output signal with the testing data.

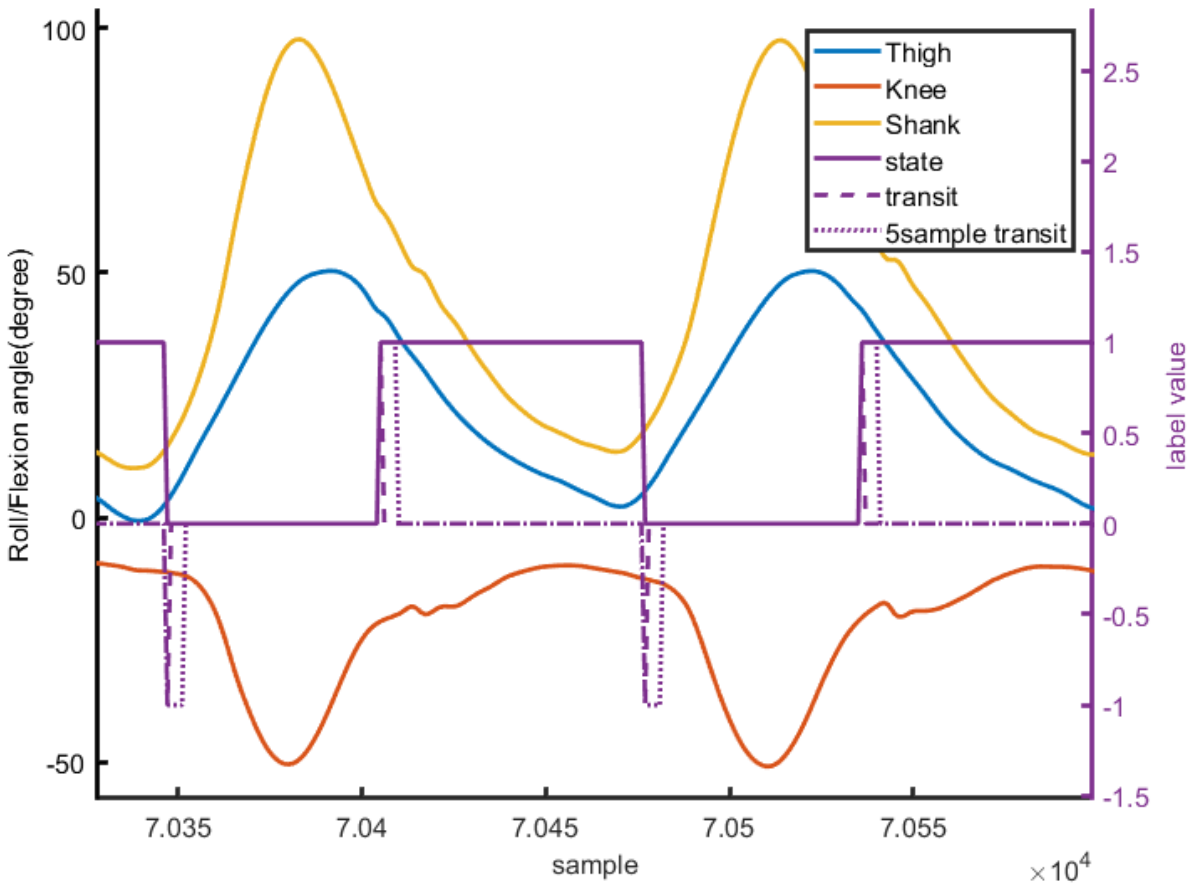


Figure 5.1: One gait cycle of the different output types with thigh roll, shank roll, and knee flexion angle.

Two machine learning approaches were chosen for it is suitable for the dataset. All ground truth of IC and FO were labelled; we would use supervised learning to train classifiers to predict the outputs. The dataset is a time-series data; therefore, we would also explore time-series neural networks.

5.2 Data Analysis

In this section of the study, we would evaluate each machine-learning technique's performance in gait phase detection consistent with chapter 4. The timing error is defined as the ground truth's timing subtract the prediction timing. A negative value indicated early detection. True positive is the nearest positive detection within 200 ms around the ground

truth; all other positive position is considered as false positives. Negative detection within 200 ms around the actual ground truth occurrence is considered a false negative. True negative is not considered, since the amount of true negative for the transition output types outnumbered the true positive for each phase.

The F1-score of each output would be computed to determine how reliable the trained models are in detecting the correct phases. Time performance is evaluated with three aspects. First is the timeliness of predicting the occurrence of each gait phase by examining the mean absolute timing error from the ground truth. The second is the consistency of predicting the gait phases using the standard deviation of the timing error. The third is the usefulness of the model for application requiring a timing error within 50 ms. The usefulness is determined by the probability of detection outside the 50 ms tolerance, given the models' mean and standard deviation across the testing data.

5.3 Time Series Neural Network

5.3.1 Training Scheme

There are two types of time series neural network structure selected for the comparison study. They are nonlinear Autoregressive with external input (NARX) and nonlinear input-output (NIO) network.

NARX network predicts the output y(t), given d past values of y(t) and another series x(t). The defining equation for the model is:

$$y(t) = f(y(t-1), \dots, y(t-d), x(t-1), \dots, x(t-d)) \tag{5.1}$$

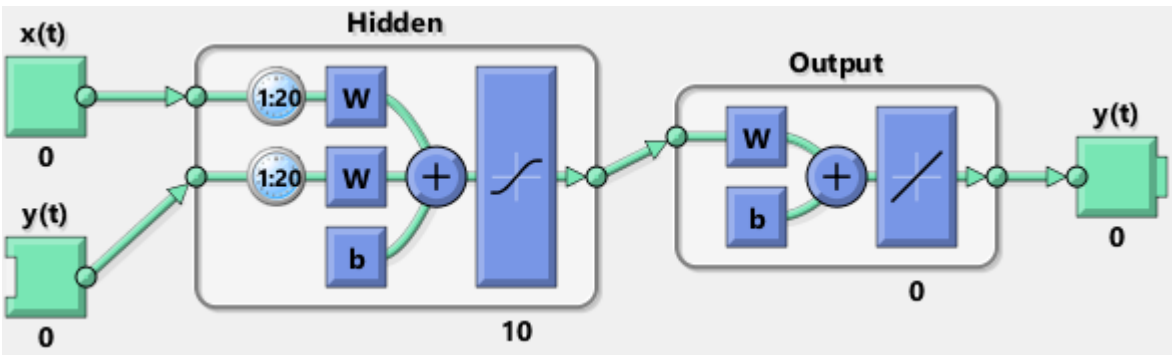


Figure 5.2: The network structure of a 20 delay and 10 neurons NARX network. This diagram is a generation with view(net) command on Matlab using Simulink diagram block.

NIO network predicts the output y(t), given d past values of x(t). The defining equation for the model is:

$$y(t) = f(x(t-1), \dots, x(t-d)) \tag{5.2}$$

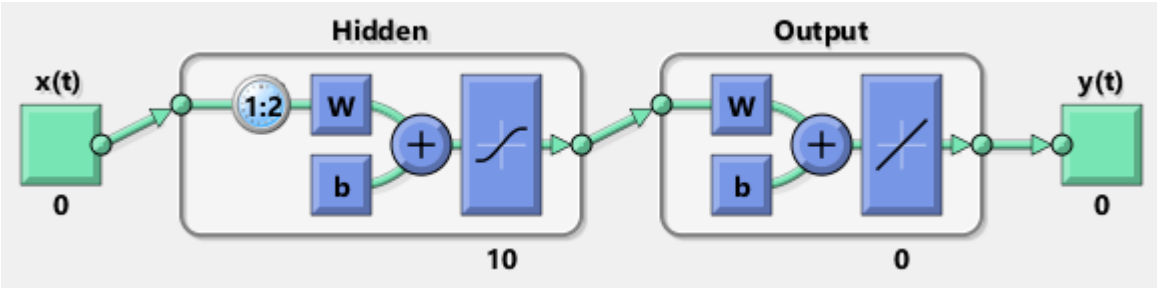


Figure 5.3: The network structure of a 2-delay and 10 neurons NIO network. This diagram is a generation with view(net) command on Matlab using the Simulink diagram block.

Each network is defined by the number of neurons, the number of delays, and the type of output. We train all networks with 10 neurons with 2 or 20 delays. The training algorithm is Levenberg-Marquardt; it has faster training time but uses more memory during training. All training is carried on a PC; thus, memory is not a limiting issue. The default setting is 6-fold cross-validation, with data distribution as follows: 70% training, 15% verify, 15% test.

The NN aims to reproduce the outcome from learning the training data. It does not restrict its prediction to be the discrete value of the training output. It is evident in the figure, and from literature [168]. An extra layer of filtering is required to rectify the raw output signal to a discrete signal that could be used in analysing the detection performance.

There are three different approaches to filtering the predicted signal from each trained NN. The first type is a latch with thresholding using hysteresis. The on/off state output has a value of 1 when on, and zero when off. The threshold is set halfway with a hysteresis of ±0.2 on each side, a high level of 0.7 and a low level of 0.3. For example, the signal above 0.7 will stay on until it falls below 0.3, and it will stay low until it went above 0.7 again. Similarly, for transition output, the threshold is halfway at zero, with a hysteresis of ±0.1. These are hysteresis values chosen after some trial and error on the training set output with an increment of 0.1 around the mid-point. The trial and error stop when the next increment does not improve the average F1-score between the two different network and the two different delays.

The second type uses the two standard deviations to be the threshold for transition output only. High level is above +2 standard deviations, and the low level is below -2 standard deviations of the training data; otherwise, it is zero. Ideally, it should be offset by the mean of the training data, which is 0.07. Given the trained data begin and end with ICs, and a random sample of 70% is taken from the original data. Therefore, the small offset in positive could be the fact that there are more ICs than FOs in training. From the actual data, the duration of the IC transition and FO transition is the same. Hence, we will assume the expected value should be zero.

The third type uses the same thresholding as the first type, but there is an 11-sample moving mean applied to the output signal before the thresholding. It is aimed to attenuate the output so that the output is minimised. Moving mean is applied only to the 5-sample wide transition output, because it is found that the attenuation is too strong for single-sample-transition output in the training data.

There is a total of 36 networks trained for both stair ascent and descent. Hereafter each trained network will be referred to by a code name of 4 numbers separated by a dash between them. Table 5.1 summarises the code name for each trained network in left to right order. The output types are on/off state (1), single sample transition output (2), 5-sample wide transition output (3). Filter types are threshold hysteresis (1), 2-sigma threshold (2), and with an 11-sample moving mean (3). The value of hysteresis is indicated by the number after the decimal place of the filter type number. For example, the filter number of 1.2 represents thresholding with a hysteresis of ±0.2, and 3.1 represents an 11-sample moving mean applied with thresholding with a hysteresis of ±0.1.

Table 5.1: A summary of all configuration of trained NN models

Code	Output type	Delay	Network type	Filter type
1	state	2	NIO	Threshold with hysteresis
2	transition	20	NARX	2-sigma
3	5-sample wide transition			11-sample Moving Mean

Filter type code +0.0 with a hysteresis of ±0.0, +0.1 with ±0.1, +0.2 with ±0.2, and so on.

5.3.2 Results

The model with the best F1-score are 1-2-1-1.3 for stair ascent IC (0.9905), 1-2-1-1.3 for stair ascent FO (0.9918), 3-2-1-3.1 for stair descent IC (0.9983), and 3-1-2-3.1 for stair descent FO (1). One model outperformed the ruled-based algorithms for stair ascent IC, 14 models for stair ascent FO, none for stair descent IC and eight models for stair descent FO outperform the rule-based algorithm in chapter 4. These models were indicated by < symbol in the table below.

The best timing error was compared and selected between the models with at least 0.9 F1-score. Among those, the timeliest (least absolute timing error) were: 1-1-1-1.1 for stair ascent IC with 30.0892 ms, 3-2-1-1.3 for stair ascent FO with 15.6488 ms, 3-1-1-1.4 for stair descent IC with 22.5235 ms, and 3-2-1-2 for stair descent FO with 16.125 ms. The most consistent (least standard deviation) were: 3-2-1-3.1 for stair ascent IC with 24.0778 ms, 3-2-1-3.1 for stair ascent FO with 22.3147 ms, 3-1-1-1.4 for stair descent IC with 29.8107 ms, and 3-2-1-2 for stair descent FO with 19.3547 ms. The most robust (least likely to have a detection outside 50 ms) were: 1-1-1-1.1 for stair ascent IC with 0.1966, 3-2-1-1.3 for stair ascent FO with 0.0324, 3-1-1-1.4 for stair descent IC with 0.0935, and 3-2-1-2 for stair descent FO with 0.014325.

Among the models that had a F1-score above 0.9, 12 models outperformed the method in chapter 4 in usefulness, 19 in timeliness and 21 in consistency for stair ascent IC, 29 in usefulness, 29 in timeliness, and none in consistency for stair ascent FO, none in usefulness, 23 in timeliness, and 1 in consistency in stair descent IC, and, none in usefulness, 19 in timeliness, and none in consistency in stair descent FO. The symbol of %, ^, and & were indicating the models that outperformed the rule-based method in timeliness, consistency, and usefulness respectively on the tables below.

Table 5.2: Time series NN performance for initial contact of stair ascent

Output	Delay	Model	Filter	TP	FP	FN	TPR	PPV	F1-score	mn(Te)	Sd(Te)	mn(|Te|)	Sd(|Te|)	pr(|Te|>50)	
1	1	1	1.3	786	14	7	0.9912	0.9825	0.9868	15.1272	41.1899	35.3562	25.9630	0.2555	%^&
		2	1.3	23	6	770	0.0290	0.7931	0.0560	-130.4348	29.6155	130.4348	29.6155	0.9967	^
	2	1	1.3	786	8	7	0.9912	0.9899	0.9905	22.8880	41.7537	38.3842	28.1544	0.2985	<%^
		2	1.3	722	146	71	0.9105	0.8318	0.8694	-128.7950	39.7993	129.1274	38.7057	0.9761	^
	1	1	1.2	785	22	8	0.9899	0.9727	0.9813	6.2803	39.2685	31.6306	24.0777	0.2087	%^&
		2	1.2	571	209	222	0.7201	0.7321	0.7260	-133.0298	38.5040	133.0298	38.5040	0.9845	^
	2	1	1.2	786	11	7	0.9912	0.9862	0.9887	13.9695	40.4877	34.4020	25.4881	0.2438	%^&
		2	1.2	716	189	77	0.9029	0.7912	0.8433	-129.2318	39.5972	129.5670	38.4847	0.9773	^
	1	1	1.1	785	36	8	0.9899	0.9562	0.9727	-0.1783	38.7228	30.0892	24.3511	0.1966	%^&
		2	1.1	537	436	256	0.6772	0.5519	0.6082	-143.4451	37.1545	143.4451	37.1545	0.9940	^
	2	1	1.1	785	16	8	0.9899	0.9800	0.9849	6.3822	38.8240	31.0701	24.1144	0.2038	%^&
		2	1.1	713	255	80	0.8991	0.7366	0.8098	-129.0042	39.5656	129.3408	38.4494	0.9771	^
2	1	1	1.0	790	4430	3	0.9962	0.1513	0.2628	-37.5570	61.0614	61.3038	37.1200	0.4951	
		2	1.0	786	6845	7	0.9912	0.1030	0.1866	-11.2723	54.4869	45.4962	31.9924	0.3690	%
	2	1	1.0	793	5615	0	1.0000	0.1238	0.2202	-24.8802	60.3861	54.9937	35.1875	0.4462	
		2	1.0	765	7101	28	0.9647	0.0973	0.1767	-42.6405	79.6809	78.9804	43.8585	0.5857	
	1	1	1.1	603	8	190	0.7604	0.9869	0.8590	-29.6186	40.6020	38.9718	31.7162	0.3328	%^
		2	1.1	753	13	40	0.9496	0.9830	0.9660	-24.5418	37.5207	33.1474	30.1777	0.2722	%^&
	2	1	1.1	777	12	16	0.9798	0.9848	0.9823	-34.0798	41.7060	43.0116	32.4028	0.3732	%^
		2	1.1	782	75	11	0.9861	0.9125	0.9479	-32.0077	39.6455	39.5013	32.1748	0.3443	%^
	1	1	1.2	3	0	790	0.0038	1.0000	0.0075	10.0000	0.0000	10.0000	0.0000	0.0000	%^&
		2	1.2	41	0	752	0.0517	1.0000	0.0983	15.8537	34.4946	26.5854	26.8896	0.1892	%^&
	2	1	1.2	40	0	753	0.0504	1.0000	0.0960	-10.2500	38.9929	33.2500	22.2327	0.2152	%^&
		2	1.2	780	25	13	0.9836	0.9689	0.9762	-14.6795	39.7460	32.7821	26.8228	0.2389	%^&
	1	1	2	737	145	56	0.9294	0.8356	0.8800	-28.4532	39.6355	36.8114	32.0115	0.3172	%^
		2	2	707	180	86	0.8916	0.7971	0.8417	-11.3296	35.4803	25.8133	26.8351	0.1798	%^&
	2	1	2	782	278	11	0.9861	0.7377	0.8440	-25.3581	39.6826	35.1023	31.3818	0.2961	%^
		2	2	595	388	198	0.7503	0.6053	0.6700	29.9160	33.9760	36.2689	27.0775	0.2866	%^&
3	1	1	1.1	705	139	88	0.8890	0.8353	0.8613	-31.8582	39.4217	38.7234	32.6929	0.3416	%^

	2	1.1	756	44	37	0.9533	0.9450	0.9492	-29.9074	39.5252	38.0820	31.7131	0.3272	%^
2	1	1.1	763	120	30	0.9622	0.8641	0.9105	-41.7431	41.1702	47.6409	34.1633	0.4335	%^
	2	1.1	784	16	9	0.9887	0.9800	0.9843	-28.9796	39.1477	38.2143	30.1867	0.3175	%^
1	1	1.2	771	31	22	0.9723	0.9613	0.9668	-22.9831	38.4123	33.4371	29.7477	0.2696	%^&
	2	1.2	331	9	462	0.4174	0.9735	0.5843	-10.1208	38.9248	31.2689	25.2423	0.2140	%^&
2	1	1.2	775	33	18	0.9773	0.9592	0.9681	-29.0839	37.4692	37.2387	29.3665	0.3057	%^
	2	1.2	766	10	27	0.9660	0.9871	0.9764	-12.2454	41.1364	33.7076	26.5456	0.2445	%^&
1	1	1.3	774	14	19	0.9760	0.9822	0.9791	-16.7054	38.0457	31.4083	27.1874	0.2305	%^&
	2	1.3	49	0	744	0.0618	1.0000	0.1164	24.8980	35.8889	34.6939	26.3076	0.2606	%^&
2	1	1.3	778	16	15	0.9811	0.9798	0.9805	-20.8612	37.3223	32.8149	27.3952	0.2463	%^&
	2	1.3	730	9	63	0.9206	0.9878	0.9530	4.2055	40.8851	32.0959	25.6465	0.2238	%^&
1	1	2	695	110	98	0.8764	0.8634	0.8698	-5.3237	37.0626	27.7410	25.1266	0.1818	%^&
	2	2	281	64	512	0.3544	0.8145	0.4938	-6.1566	36.9676	28.3630	24.4399	0.1822	%^&
2	1	2	774	171	19	0.9760	0.8190	0.8907	-1.4987	34.9914	26.3307	23.0748	0.1534	%^&
	2	2	578	252	215	0.7289	0.6964	0.7123	39.0311	39.0231	44.1869	33.0607	0.4006	%^
1	1	3.1	774	24	19	0.9760	0.9699	0.9730	-54.1473	37.2327	55.4134	35.3181	0.5469	^
	2	3.1	0	0	793			0.0000					1.0000	
2	1	3.1	776	18	17	0.9786	0.9773	0.9779	-60.6443	36.7499	61.8557	34.6695	0.6153	^
	2	3.1	609	488	184	0.7680	0.5552	0.6444	-128.9163	38.6168	128.9491	38.5068	0.9795	^
1	1	3.0	465	1859	328	0.5864	0.2001	0.2984	-89.8280	76.2056	107.5914	47.8842	0.7326	
	2	3.0	0	0	793			0.0000					1.0000	
2	1	3.0	640	1629	153	0.8071	0.2821	0.4180	-90.4063	74.7735	109.2188	42.7756	0.7357	
	2	3.0	611	791	182	0.7705	0.4358	0.5567	-137.0704	37.7884	137.0704	37.7884	0.9894	^

mn(Te) and sd(Te) represents the mean of timing error and the standard deviation of the timing error, respectively. Similarly, the next two columns are for the absolute timing error, |Te|. Pr(|Te|>50) is the likelihood for detection to have a fundamental timing error greater than 50 ms. Models that outperform the rule-based approach in chapter 4 are indicated with the symbol < for F1-score, % for the mean of absolute timing error, ^ for the standard deviation of timing error, and & for the likelihood to have a |T2|> 50 ms

Table 5.3: Time series NN performance for end contact of stair ascent

Output	Delay	Model	Filter	TP	FP	FN	TPR	PPV	F1-score	mn(Te)	Sd(Te)	mn(|Te|)	Sd(|Te|)	pr(|Te|>50)	
1	1	1	1.3	788	13	6	0.9924	0.9838	0.9881	21.1675	26.4773	24.8223	23.0814	0.1417	<%&
		2	1.3	12	17	782	0.0151	0.4138	0.0292	15.0000	19.3061	18.3333	15.8592	0.0353	%^&
	2	1	1.3	788	7	6	0.9924	0.9912	0.9918	22.8553	26.0942	26.0787	22.8687	0.1517	<%&
		2	1.3	757	111	37	0.9534	0.8721	0.9110	33.8045	33.1551	36.4465	30.2227	0.3183	%&
	1	1	1.2	789	19	5	0.9937	0.9765	0.9850	15.8809	26.5607	21.3054	22.4387	0.1060	<%&
		2	1.2	704	76	90	0.8866	0.9026	0.8945	39.3466	33.2177	40.7955	31.4187	0.3778	%
	2	1	1.2	788	10	6	0.9924	0.9875	0.9899	17.1954	24.8477	21.4848	21.2435	0.0968	<%&
		2	1.2	770	135	24	0.9698	0.8508	0.9064	26.7273	32.4739	29.9740	29.4995	0.2459	%&
	1	1	1.1	788	34	6	0.9924	0.9586	0.9752	10.5584	24.7106	18.3756	19.5996	0.0624	%&
		2	1.1	765	208	29	0.9635	0.7862	0.8659	29.8693	30.6685	32.3791	28.0022	0.2604	%&
	2	1	1.1	788	14	6	0.9924	0.9825	0.9875	11.8274	24.5718	18.6294	19.9083	0.0661	<%&
		2	1.1	777	191	17	0.9786	0.8027	0.8820	21.9048	30.1795	25.3024	27.3897	0.1845	%&
2	1	1	1.0	775	4445	19	0.9761	0.1485	0.2577	-52.5806	67.6692	79.0839	32.9405	0.5800	<
		2	1.0	794	6837	0	1.0000	0.1040	0.1885	-33.0730	57.1198	60.1763	27.0578	0.4564	<
	2	1	1.0	745	5662	49	0.9383	0.1163	0.2069	-17.1946	91.6004	78.5906	50.0191	0.5917	<
		2	1.0	744	7121	50	0.9370	0.0946	0.1718	-74.8387	74.8420	92.5000	51.3997	0.6777	<
	1	1	1.1	607	5	187	0.7645	0.9918	0.8634	-15.0082	25.5105	22.0264	19.7596	0.0905	<%&
		2	1.1	755	12	39	0.9509	0.9844	0.9673	-17.2185	24.9764	23.4967	19.1798	0.0982	<%&
	2	1	1.1	779	11	15	0.9811	0.9861	0.9836	-8.2798	23.6212	17.6765	17.7129	0.0455	%&
		2	1.1	714	144	80	0.8992	0.8322	0.8644	-15.4622	25.6991	23.5854	18.5150	0.0949	%&
	1	1	1.2	4	0	790	0.0050	1.0000	0.0100	-22.5000	55.6028	42.5000	36.8556	0.4066	%
		2	1.2	42	0	752	0.0529	1.0000	0.1005	-4.2857	37.1643	19.0476	32.0677	0.1814	%&
	2	1	1.2	40	1	754	0.0504	0.9756	0.0958	4.2500	17.0801	13.2500	11.4102	0.0044	%^&
		2	1.2	763	43	31	0.9610	0.9467	0.9538	0.1573	27.3353	19.0301	19.6118	0.0674	%&
	1	1	2	789	28	5	0.9937	0.9657	0.9795	-17.1863	26.3603	23.7262	20.6630	0.1120	%&
		2	2	789	92	5	0.9937	0.8956	0.9421	-13.4221	25.7714	21.6096	19.4158	0.0848	%&
	2	1	2	787	31	7	0.9912	0.9621	0.9764	-8.4371	23.5408	17.6112	17.7453	0.0453	%&
		2	2	766	82	28	0.9647	0.9033	0.9330	21.6449	26.6453	24.7520	23.7829	0.1472	%&
3	1	1	1.1	767	77	27	0.9660	0.9088	0.9365	-21.7992	27.8357	26.8318	23.0167	0.1605	%&

	2	1.1	779	22	15	0.9811	0.9725	0.9768	-13.9795	25.6450	21.9384	19.2725	0.0864	%&
2	1	1.1	627	256	167	0.7897	0.7101	0.7478	-17.3525	26.4758	23.6045	21.0831	0.1142	%&
	2	1.1	778	23	16	0.9798	0.9713	0.9755	-16.6324	24.3731	22.4422	19.1506	0.0886	%&
1	1	1.2	787	15	7	0.9912	0.9813	0.9862	-14.2440	25.7366	21.8933	19.6366	0.0886	<%&
	2	1.2	336	5	458	0.4232	0.9853	0.5921	-4.6726	20.2495	15.7440	13.5398	0.0161	%&
2	1	1.2	723	85	71	0.9106	0.8948	0.9026	-6.8880	23.9777	17.5657	17.7047	0.0449	%&
	2	1.2	766	10	27	0.9660	0.9871	0.9764	-12.2454	41.1364	33.7076	26.5456	0.2445	%&
1	1	1.3	781	8	13	0.9836	0.9899	0.9867	-9.1037	25.7496	19.9616	18.6293	0.0670	<%&
	2	1.3	49	1	745	0.0617	0.9800	0.1161	8.3673	23.3940	17.3469	17.6512	0.0439	%&
2	1	1.3	763	32	31	0.9610	0.9597	0.9604	-0.6553	23.3560	15.6488	17.3415	0.0324	%&
	2	1.3	734	6	60	0.9244	0.9919	0.9570	10.8038	23.9550	17.1798	19.8790	0.0565	%&
1	1	2	785	79	9	0.9887	0.9086	0.9469	-1.4013	25.6064	18.7261	17.5082	0.0512	%&
	2	2	783	87	11	0.9861	0.9000	0.9411	-0.3959	24.8416	17.5862	17.5383	0.0442	%&
2	1	2	785	25	9	0.9887	0.9691	0.9788	7.5796	23.5769	16.2930	18.6438	0.0433	%&
	2	2	764	94	30	0.9622	0.8904	0.9249	20.5890	25.7886	23.4686	23.1952	0.1301	%&
1	1	3.1	786	12	8	0.9899	0.9850	0.9874	-46.6285	24.6289	48.0025	21.8263	0.4456	<
	2	3.1	0	0	794	0.0000		0.0000					1.0000	
2	1	3.1	778	16	16	0.9798	0.9798	0.9798	-40.6170	22.3147	42.4936	18.4873	0.3371	%&
	2	3.1	459	638	335	0.5781	0.4184	0.4855	27.4728	44.9956	33.1373	40.9943	0.3509	%&
1	1	3.0	725	1599	69	0.9131	0.3120	0.4650	-76.2207	42.2575	80.9655	32.2311	0.7339	
	2	3.0	0	0	794	0.0000		0.0000					1.0000	
2	1	3.0	660	1609	134	0.8312	0.2909	0.4310	-67.5909	69.5604	87.5000	41.7881	0.6453	
	2	3.0	672	730	122	0.8463	0.4793	0.6120	15.5208	33.3571	25.4315	26.5748	0.1754	%&

mn(Te) and sd(Te) represents the mean of timing error and the standard deviation of the timing error, respectively. Similarly, the next two columns are for the absolute timing error, |Te|. Pr(|Te|>50) is the likelihood for detection to have a fundamental timing error greater than 50 ms. Models which outperform the rule-based approach in chapter 4 are indicated with the symbol < for F1-score, % for the mean of absolute timing error, ^ for the standard deviation of timing error, and & for the likelihood to have a |T2|> 50 ms

Table 5.4: Time series NN performance for initial contact of stair descent

Output	Delay	Model	Filter	TP	FP	FN	TPR	PPV	F1-score	mn(Te)	Sd(Te)	mn(|Te|)	Sd(|Te|)	pr(|Te|>50)	
1	1	1	1.4	879	5	0	1	0.9943	0.9972	37.3265	30.6501	41.0125	25.5018	0.3418	
		2	1.4	708	170	171	0.8055	0.8064	0.8059	-132.119	40.8645	132.1186	40.8645	0.9778	
	2	1	1.4	879	6	0	1	0.9932	0.9966	35.4152	30.5609	39.8749	24.4504	0.3192	
		2	1.4	838	9	41	0.9534	0.9894	0.971	17.9594	62.9541	54.2601	36.585	0.4456	
	1	1	1.3	879	10	0	1	0.9888	0.9943	29.124	32.2544	35.9954	24.3387	0.2658	%
		2	1.3	674	205	205	0.7668	0.7668	0.7668	-135.623	38.023	135.6231	38.023	0.9878	
	2	1	1.3	879	12	0	1	0.9865	0.9932	25.5176	31.212	33.2765	22.7486	0.2242	%
		2	1.3	855	38	24	0.9727	0.9574	0.965	-32.2573	76.5881	66.3158	50.0456	0.5498	
	1	1	0.2	879	25	0	1	0.9723	0.986	21.3993	33.3645	31.7975	23.6518	0.2118	%
		2	0.2	656	223	223	0.7463	0.7463	0.7463	-159.638	67.9488	161.8553	62.4776	0.9477	
	2	1	0.2	879	23	0	1	0.9745	0.9871	18.7713	31.7929	30.1479	21.2984	0.1783	%
		2	0.2	842	113	37	0.9579	0.8817	0.9182	-56.8409	70.5221	72.0428	54.8795	0.6035	
	1	1	1.1	879	52	0	1	0.9441	0.9713	14.3117	33.531	28.3732	22.8786	0.1711	%
		2	1.1	640	241	239	0.7281	0.7264	0.7273	-141.797	35.3749	141.7969	35.3749	0.9953	
	2	1	1.1	879	37	0	1	0.9596	0.9794	12.6507	33.1615	28.1911	21.5468	0.1595	%
		2	1.1	825	393	54	0.9386	0.6773	0.7868	-55.8424	64.9163	66.0485	54.4848	0.5874	
	1	1	1	879	149	0	1	0.8551	0.9219	8.1229	34.014	26.7577	22.4993	0.1529	%
		2	1	627	275	252	0.7133	0.6951	0.7041	-144.849	34.5269	144.8485	34.5269	0.997	
	2	1	1	879	102	0	1	0.896	0.9452	7.8043	34.315	27.4175	22.0436	0.1555	%
		2	1	815	930	64	0.9272	0.467	0.6212	-39.9877	56.6998	49.4847	48.6221	0.4862	
2	1	1	1	879	6741	0	1	0.1154	0.2068	-6.6439	67.6804	58.5666	34.5091	0.4622	
		2	1	879	7933	0	1	0.0998	0.1814	-6.1661	54.8635	45.7338	30.8885	0.3651	
	2	1	1	879	8394	0	1	0.0948	0.1732	-17.1331	59.2491	51.9681	33.1747	0.4181	
		2	1	871	4685	8	0.9909	0.1568	0.2707	-50.31	65.3568	68.5419	45.8489	0.5643	
	1	1	1.1	830	1	49	0.9443	0.9988	0.9708	-11.6506	36.6188	27.8193	26.4952	0.1936	%
		2	1.1	864	43	15	0.9829	0.9526	0.9675	-14.0625	38.5851	29.3171	28.7454	0.2243	%
	2	1	1.1	730	3	149	0.8305	0.9959	0.9057	-14.6438	47.2706	38.5342	31.0213	0.313	
		2	1.1	879	231	0	1	0.7919	0.8839	-13.8794	44.2949	35.6542	29.7024	0.282	%
	1	1	1.2	181	0	698	0.2059	1	0.3415	6.1878	31.6113	25.7459	19.2677	0.1206	%

		2	1.2	686	2	193	0.7804	0.9971	0.8756	5.5831	30.5979	23.6297	20.2056	0.1079	%
	2	1	1.2	217	0	662	0.2469	1	0.396	13.8249	41.9284	33.4562	28.7315	0.2581	%
		2	1.2	870	172	9	0.9898	0.8349	0.9058	1.6897	41.8064	32.3103	26.5607	0.2321	%
	1	1	2	828	331	51	0.942	0.7144	0.8126	-7.4396	34.1023	25.0242	24.319	0.1521	%
		2	2	707	294	172	0.8043	0.7063	0.7521	6.3932	29.8794	22.3479	20.8224	0.1018	%^
	2	1	2	712	47	167	0.81	0.9381	0.8694	-12.2753	46.0098	37.0225	29.9199	0.2941	
		2	2	531	441	348	0.6041	0.5463	0.5737	56.4595	36.5272	58.1921	33.693	0.572	
3	1	1	1.1	788	388	91	0.8965	0.6701	0.7669	-48.2868	77.3808	69.3528	59.2152	0.5932	
		2	1.1	871	9	8	0.9909	0.9898	0.9903	-20.1148	41.9111	33.2721	32.4546	0.2851	%
	2	1	1.1	824	436	55	0.9374	0.654	0.7705	-23.6893	61.3793	47.9612	45.0135	0.449	
		2	1.1	879	203	0	1	0.8124	0.8965	-13.0262	37.6707	29.6587	26.6141	0.2103	%
	1	1	1.2	874	24	5	0.9943	0.9733	0.9837	-22.6659	44.4347	35.4119	35.1189	0.3202	%
		2	1.2	821	1	58	0.934	0.9988	0.9653	-2.7162	31.7237	23.3252	21.6576	0.1163	%
	2	1	1.2	851	112	28	0.9681	0.8837	0.924	-16.7568	45.9264	35.2056	33.9037	0.3076	%
		2	1.2	879	78	0	1	0.9185	0.9575	-1.1718	34.5273	26.1547	22.5535	0.1478	%
	1	1	1.3	876	0	3	0.9966	1	0.9983	-8.9155	35.7047	26.3813	25.6444	0.1744	%
		2	1.3	690	0	189	0.785	1	0.8795	7.7971	26.3903	21.942	16.5885	0.0691	%^&
	2	1	1.3	876	32	3	0.9966	0.9648	0.9804	-8.0137	40.5807	30.6393	27.7714	0.2268	%
		2	1.3	868	34	11	0.9875	0.9623	0.9747	8.6406	32.5147	26.1751	21.1192	0.1373	%
	1	1	1.4	852	0	27	0.9693	1	0.9844	-0.1995	29.8107	22.5235	19.5145	0.0935	%^
		2	1.4	376	0	503	0.4278	1	0.5992	13.1649	22.5881	21.9947	14.1047	0.0541	%^&
	2	1	1.4	875	7	4	0.9954	0.9921	0.9938	-1.6914	37.9735	28.8686	24.7084	0.1884	%
		2	1.4	847	16	32	0.9636	0.9815	0.9724	17.7568	32.3642	29.2326	22.5292	0.1777	%
	1	1	2	834	278	45	0.9488	0.75	0.8378	5.6835	25.8321	20.1439	17.1282	0.0587	%^&
		2	2	801	216	78	0.9113	0.7876	0.8449	2.3471	28.2398	21.3983	18.5619	0.0777	%^&
	2	1	2	868	155	11	0.9875	0.8485	0.9127	4.4816	34.7286	26.947	22.3429	0.1533	%
		2	2	727	470	152	0.8271	0.6074	0.7004	33.6039	31.2891	36.9326	27.2742	0.3039	
	1	1	3.1	876	2	3	0.9966	0.9977	0.9972	-52.1119	35.7834	52.4543	35.2789	0.5257	
		2	3.1	341	539	538	0.3879	0.3875	0.3877	-166.598	24.1475	166.5982	24.1475	1	^
	2	1	3.1	878	2	1	0.9989	0.9977	0.9983	-48.8952	40.9108	50.6492	38.7156	0.497	
		2	3.1	0	0	879	0	0	0					1	

1	1	3	804	2107	75	0.9147	0.2762	0.4243	-78.6816	93.4319	112.1642	48.2861	0.7048	
	2	3	262	625	617	0.2981	0.2954	0.2967	-171.641	40.8058	175.1527	21.0404	0.9986	
2	1	3	805	1794	74	0.9158	0.3097	0.4629	-64.646	80.649	89.7143	51.2825	0.6496	
	2	3	84	797	795	0.0956	0.0953	0.0955	-187.024	13.4236	187.0238	13.4236	1	^

mn(Te) and sd(Te) represents the mean of timing error and the standard deviation of the timing error, respectively. Similarly, the next two columns are for the absolute timing error, |Te|. Pr(|Te|>50) is the likelihood for detection to have a fundamental timing error greater than 50 ms. Models which outperform the rule-based approach in chapter 4 are indicated with the symbol < for F1-score, % for the mean of absolute timing error, ^ for the standard deviation of timing error, and & for the likelihood to have a |T2|> 50 ms

Table 5.5: Time series NN performance for end contact of stair descent

Output	Delay	Model	Filter	TP	FP	FN	TPR	PPV	F1-score	mn(Te)	Sd(Te)	mn(|Te|)	Sd(|Te|)	pr(|Te|>50)	
1	1	1	1.4	880	5	0	1	0.9944	0.9972	27.7159	20.9150	29.7159	17.9567	0.1434	<
		2	1.4	878	0	2	0.9977	1.0000	0.9989	58.6788	23.3673	58.8155	23.0207	0.6448	<
	2	1	1.4	880	6	0	1	0.9932	0.9966	32.3864	22.1790	33.2273	20.8967	0.2137	<
		2	1.4	844	4	36	0.9590	0.9953	0.9769	75.3318	29.5359	75.3555	29.4753	0.8045	
	1	1	1.3	880	10	0	1	0.9888	0.9944	18.3182	20.6579	22.9318	15.3694	0.0630	
		2	1.3	879	0	1	0.9989	1.0000	0.9994	52.5939	24.1645	52.8669	23.5605	0.5428	<
	2	1	1.3	880	12	0	1	0.9865	0.9932	26.4773	22.1249	28.0227	20.1293	0.1441	
		2	1.3	879	15	1	0.9989	0.9832	0.9910	53.7656	25.8838	53.9249	25.5500	0.5579	
	1	1	0.2	880	25	0	1	0.9724	0.9860	10.9659	21.0714	19.1477	14.0477	0.0339	%
		2	0.2	879	0	1	0.9989	1.0000	0.9994	46.1547	24.6341	46.6098	23.7608	0.4380	<
	2	1	0.2	880	23	0	1.0000	0.9745	0.9871	20.8182	21.2095	23.5909	18.0712	0.0848	
		2	0.2	880	76	0	1.0000	0.9205	0.9586	42.3977	22.7626	42.7386	22.1151	0.3692	
	1	1	1.1	880	52	0	1.0000	0.9442	0.9713	4.8295	21.3688	17.1023	13.6805	0.0224	%
		2	1.1	880	1	0	1.0000	0.9989	0.9994	39.6477	24.9503	40.3295	23.8311	0.3393	<
	2	1	1.1	880	37	0	1.0000	0.9597	0.9794	16.1364	20.6400	20.7273	16.0181	0.0511	%
		2	1.1	880	338	0	1.0000	0.7225	0.8389	33.4545	21.0070	34.0455	20.0340	0.2155	
	1	1	1	880	149	0	1.0000	0.8552	0.9219	-0.9659	21.8611	16.6477	14.1908	0.0223	%
		2	1	880	22	0	1.0000	0.9756	0.9877	31.6023	24.4866	33.0341	22.5153	0.2267	
	2	1	1	880	102	0	1.0000	0.8961	0.9452	12.2614	20.8521	18.6705	15.3734	0.0366	%
		2	1	880	865	0	1.0000	0.5043	0.6705	24.7159	19.4676	25.9659	17.7641	0.0971	
2	1	1	1	879	6741	1	0.9989	0.1154	0.2068	-50.9898	50.0232	60.8419	37.4071	0.5296	
		2	1	830	7982	50	0.9432	0.0942	0.1713	-28.0843	87.4726	78.8313	47.1090	0.5871	
	2	1	1	873	8400	7	0.9920	0.0941	0.1720	13.7801	94.9440	88.3505	37.2788	0.6023	
		2	1	844	4712	36	0.9591	0.1519	0.2623	-53.7322	59.4772	65.8886	45.6251	0.5656	
	1	1	1.1	793	39	87	0.9011	0.9531	0.9264	-17.3140	23.1371	22.5347	18.0839	0.0807	
		2	1.1	829	79	51	0.9420	0.9130	0.9273	-11.9180	23.6738	20.8203	16.3905	0.0583	%
	2	1	1.1	707	27	173	0.8034	0.9632	0.8761	-8.6139	23.1708	16.9873	17.9503	0.0427	%
		2	1.1	880	231	0	1.0000	0.7921	0.8840	-15.1932	24.4755	22.7159	17.7073	0.0814	
	1	1	1.2	182	0	698	0.2068	1.0000	0.3427	-7.4725	19.9221	16.8132	12.9921	0.0184	%

		2	1.2	682	7	198	0.7750	0.9898	0.8693	-2.3314	21.4590	16.8475	13.4788	0.0205	%
	2	1	1.2	217	1	663	0.2466	0.9954	0.3953	6.6359	15.1908	13.2719	9.9016	0.0023	%^&
		2	1.2	871	172	9	0.9898	0.8351	0.9059	-1.0563	22.8797	18.1860	13.9100	0.0290	%
	1	1	2	880	81	0	1.0000	0.9157	0.9560	-16.8750	22.7042	22.1705	17.5636	0.0739	
		2	2	876	33	4	0.9955	0.9637	0.9793	-2.9110	21.7959	17.0890	13.8266	0.0230	%
	2	1	2	880	145	0	1.0000	0.8585	0.9239	-5.3068	20.6250	16.2386	13.7692	0.0188	%
		2	2	766	144	114	0.8705	0.8418	0.8559	57.4021	25.9192	57.7154	25.2129	0.6124	
3	1	1	1.1	637	540	243	0.7239	0.5412	0.6193	-34.0345	46.1698	45.4003	35.0342	0.3991	
		2	1.1	776	105	104	0.8818	0.8808	0.8813	-18.4149	21.5631	21.9716	17.9204	0.0722	
	2	1	1.1	721	540	159	0.8193	0.5718	0.6735	-40.3329	37.4781	42.1914	35.3698	0.4062	
		2	1.1	846	237	34	0.9614	0.7812	0.8619	-24.2908	26.7295	28.3570	22.3638	0.1708	
	1	1	1.2	780	119	100	0.8864	0.8676	0.8769	-27.6026	25.4060	29.7821	22.8087	0.1901	
		2	1.2	805	18	75	0.9148	0.9781	0.9454	4.6460	22.0815	17.5652	14.1522	0.0267	%
	2	1	1.2	793	171	87	0.9011	0.8226	0.8601	-10.5549	29.9991	19.9874	24.7283	0.1160	%
		2	1.2	869	89	11	0.9875	0.9071	0.9456	-5.6732	24.2522	18.9528	16.1483	0.0446	%
	1	1	1.3	809	68	71	0.9193	0.9225	0.9209	-11.4215	24.8626	21.0630	17.4517	0.0671	%
		2	1.3	690	1	190	0.7841	0.9986	0.8784	15.6667	24.4458	22.6812	18.1167	0.0837	
	2	1	1.3	827	82	53	0.9398	0.9098	0.9245	-2.5030	23.9683	16.6022	17.4581	0.0380	%
		2	1.3	865	38	15	0.9830	0.9579	0.9703	6.7399	22.2519	19.0173	13.3622	0.0313	%
	1	1	1.4	810	43	70	0.9205	0.9496	0.9348	2.5432	23.2565	18.7160	14.0221	0.0326	%
		2	1.4	377	0	503	0.4284	1.0000	0.5998	19.7613	21.4921	23.5279	17.2750	0.0803	
	2	1	1.4	839	44	41	0.9534	0.9502	0.9518	2.6579	22.6779	16.2217	16.0594	0.0285	%
		2	1.4	841	23	39	0.9557	0.9734	0.9644	17.8240	21.2502	23.1034	15.3375	0.0657	
	1	1	2	876	107	4	0.9955	0.8911	0.9404	7.4543	21.7840	18.2534	14.0215	0.0296	%
		2	2	878	24	2	0.9977	0.9734	0.9854	7.9727	22.9667	18.6560	15.5776	0.0394	%
	2	1	2	880	40	0	1.0000	0.9565	0.9778	6.6932	19.3547	16.1250	12.6152	0.0143	%
		2	2	783	80	97	0.8898	0.9073	0.8985	51.0345	30.9331	51.8774	29.4958	0.5139	
	1	1	3.1	855	24	25	0.9716	0.9727	0.9721	-54.7836	24.6173	54.8304	24.5128	0.4085	
		2	3.1	880	0	0	1.0000	1.0000	1.0000	-36.9432	23.8090	37.8523	22.3337	0.2918	<
	2	1	3.1	844	37	36	0.9591	0.9580	0.9585	-46.9076	22.4435	47.0972	22.0424	0.3412	
		2	3.1	0	0	880	0.0000	0.0000	0.0000	0.0000				1.0000	

1	1	3	669	2242	211	0.7602	0.2298	0.3529	-62.7952	103.3545	113.2287	42.3244	0.7691	
	2	3	880	7	0	1.0000	0.9921	0.9960	-53.0000	25.0978	53.1591	24.7586	0.3986	<
2	1	3	623	1977	257	0.7080	0.2396	0.3580	-87.4318	108.3143	131.7978	44.6095	0.7858	
	2	3	53	828	827	0.0602	0.0602	0.0602	183.5849	26.6093	183.5849	26.6093	0.7875	

mn(Te) and sd(Te) represents the mean of timing error and the standard deviation of the timing error, respectively. Similarly, the next two columns are for the absolute timing error, |Te|. Pr(|Te|>50) is the likelihood for detection to have a fundamental timing error greater than 50 ms. Models which outperform the rule-based approach in chapter 4 are indicated with the symbol < for F1-score, % for the mean of absolute timing error, ^ for the standard deviation of timing error, and & for the likelihood to have a |T2|> 50 ms

5.3.3 Discussion

Results align with the hypothesis of this chapter; some neural networks do offer better timing performance in phases where rule-based methods are difficult to define, especially for stair ascent, where more networks outperform the rule-based approach than stair descent. It is possible to implement these networks to supplement the performance of the rule-based.

It also aligns with our expectation that transition output tends to have better timing consistency than on/off state output. Table 5.6 summarises the average value of each aspect of timing performance from the models with the same output type with at least 0.9 in F1-score. Except for stair ascent IC, trained models using transition output have better timing performance in all three aspects than state output. It is expected that stair ascent IC to have worse performance because it is also the most inconsistent events from the observation made in Chapter 4. The enlarged error for transition output could be overshadowed by the physical inconsistency of the event occurrence.

Table 5.6: Average timing performance of the model with F1-score above 0.9

			Output type		
Activity	Event	average()	1	2	3
Stair Ascent	Initial contact	mn(|Te|)	33.4887	37.1106	40.1735
		sd(Te)	40.0411	39.6545	38.8270
		pr(|Te|>50)	0.2345	0.3072	0.3419
	Foot off	mn(|Te|)	24.6396	21.1289	24.2493
		sd(Te)	27.3614	25.4644	25.9015
		pr(|Te|>50)	0.1486	0.0858	0.1281
Stair Descent	Initial contact	mn(|Te|)	39.6510	31.9952	31.9457
		sd(Te)	41.1477	41.0702	37.0639
		pr(|Te|>50)	0.2907	0.2407	0.2394
	Foot off	mn(|Te|)	35.1922	19.5065	26.4810
		sd(Te)	22.9258	22.4693	22.9783
		pr(|Te|>50)	0.2632	0.0473	0.1239

mn(Te) and sd(Te) represents the mean of timing error and the standard deviation of the timing error, respectively. Pr(|Te|>50) is the likelihood for detection to have a fundamental timing error greater than 50 ms.

Trained networks typically have similar performance between IC and FO. The networks are trained with both IC and FO outputs, whereas the rule-based method in Chapter 4 is having a specific requirement for each particular phase. It indicates that neural networks have better generalisation for detection gait phases, whereas rule-based approach depends on the quality of the definition for each phase.

State output networks tend to give better F1-score, whereas transition output gives better timing performance. This observation aligns with the hypothesis that using the transition as output instead of the original on/off state output will predict those transitions closer to the actual truth. However, the trade-off of this timing accuracy reduces its ability to correctly detect the transition output as reflected by the reduction in F1-score. This trade-off could be interpreted that the networks are trained with a narrower range for each output state; hence any prediction made for those outputs would have a smaller deviation from the truth. The reduction of this output size also transfers to the increased likelihood of a missed prediction. The trade-off between correct detection and timing performance using a modified output signal would be an option for developers depending on their application requirements.

The effects of the delay have little relationship with the performance. In general, NARX has a better F1-score with 2 delays than 20 delays. However, the opposite is true for those 2-delay networks that already have very poor scores. The effect of having more delay is more consistent among NIO networks, where higher delay produces better F1-score. The instability of the NARX could be due to the feedback of the predicted output. The feedback may reinforce either an error or a correction.

The filtering type is a layer of design to be considered. It is difficult to decide which filtering is best suited for the type of network or the output type. In this study, trial and error are used to find a more suitable thresholding value, with an even hysteresis spread from the mid-point of the training output range. This process can be improved by using an optimisation process to find the optimal thresholding level for each discrete output level. However, this is not within the study's scope, where we aimed to find a readily available network type that can improve or complement the rule-based algorithm in Chapter 4. Nevertheless, the filtering of the predicted outcome should be integrated as a part of the network training in a future study.

Moving mean filters worsen the detection in both F1-score and timing error. Despite the smoothening of the signal, the attenuation of the signal makes the thresholding more difficult to distinguish a positive from a negative detection. Moving mean transformation always shrink the period of non-zero output; this negatively affected the timing performance.

Neural networks are an active research field. There could be other useful techniques, especially those in the field of computer vision. It may be a research direction to establish a systematically procedure to transform time-series data into a pseudo-image for more advanced networks to predict the gait phases. Some researchers have attempted such processing in other fields [169, 170].

One of the purposes of using a neural network is to have one single model that can determine both IC and FO occurrence. In this study, the network has one output, and we use a different output value to represent the different labels. In theory, we could have more output value representing the gait progression [168]. If more gait phases are included, the output has to be signal sides (all positive output, or negative output), and the order of occurrence has to be in increasing or decreasing order so that it would form a 'saw-tooth' shape. When this approach is used, the output type that specifically labels each phase's transition cannot be used because the region between each transition will share the same output value. The resultant waveform will be spikes or pillars with different heights along a datum line. This waveform will cause an ambiguity issue when the output is increasing to a taller spike. Before the spike reaches its full height, it would reach a lower value that is recognised as another phase transition. In this situation, the analysis and classification of each gait phase have to be done after the prediction spike returns to the datum line. This will cause additional delay to an already time-critical application. As indicated by the results in this study, using transition only output does improve the timing performance of the detection. However, a knowledge gap is present of how best to utilise this type of output signal.

This study has made it clear about some of the downfalls when training neural networks for gait data. It also offers suggestions on how different existing and emerging techniques could be incorporated for future follow-up studies.

5.4 Supervised Learning

5.4.1 Training scheme

Training a model in MATLAB consisted of two parts: 1) the validated model that is trained with the validation scheme in our study it would be 6-fold cross-validation; 2) the full model that is trained with all data. MATLAB has different minimisation functions for different

classifier types, and each model would be trained until the optimiser cannot significantly improve the result.

We compare the performance of all available classifier in MATLAB with the same training data as the NN section to compare the performance of the two machine learning methods in this study. There are also three different output types. The classifier will be subject to the same evaluation procedure as the NN and the adaptive statistical ruled-based method in Chapter 4.

Table 5.7: A summary of all configuration of trained models

Code	Output type	Cost type	Model type
1	state	equal cost	Fine tree
2	transition	transition 5, steady 1	Medium tree
3	5-sample wide transition		Coarse tree
4			Linear Discriminant
5			Quadratic Discriminant
6			Logistic Regression
7			Gaussian Naïve Bayes
8			Kernal Naïve Bayes
9			Linear SVM
10			Quadratic SVM
11			Cubic SVM
12			Fine SVM
13			Medium SVM
14			Coarse SVM
15			Fine KNN
16			Medium KNN
17			Coarse KNN
18			Cosine KNN
19			Cubic KNN
20			Weighted KNN
21			Boosted Trees
22			Bagged Trees
23			Subspace Discriminant
24			Subspace KNN
25			RUSBoosted Trees

The default cost matrix is an evenly distributed cost for all output state. A modified cost matrix is applied to two transition output types. In the modified cost matrix, the cost of misclassifying the rising and falling edge output is five times higher than the steady-state. It aims to maximise the true positive classification of the transitional edges. Some cubic SVM models are difficult to train and didn't yield a good accuracy in the training; therefore, we will also discard some combination with Cubic SVM. Some model types are not suitable for some output type. For example, Logistic Regression only trains output with binary value; therefore, it cannot use with the transition output type. A total of 118 classifiers are trained for stair ascent, and 119 classifiers for stair descent.

All classifier hereafter is referred by their code name for simplicity in writing. The code name is formatted with three numbers separated by the dash symbol, each number representing the output, misclassification, and model type of the classifier. For example, a codename of 3-2-2 classifier is referring to medium decision tree model for 5-sample wide transition output with transition output cost five times of the steady-state. Table 5.7 lists the code of each output, cost, and model type trained in this study.

5.4.2 Results

Trained models that failed to respond to any input at all are excluded from the table. They are considered a failure because they did not change its output signal throughout the test data. These models are trained to maximise the prediction of true negative detection, which are the steady-state output in the transition output type.

The classifiers with the best F1-score were 1-1-23 for stair ascent IC (0.9631), 3-2-21 for Stair ascent FO (0.9887), and 1-1-14 for stair descent IC (0.9943). Multiple classifiers had achieved a perfect F1-score of 1 for stair descent FO. They were: 2-1-25, 2-2-7, 2-2-8, 3-1-7, 3-1-8, 3-2-3, 3-2-7, and 3-2-8. This aligned with the expectation. The same trend had been observed between the timing consistency of FO and IC, and between stair descent and stair ascent during the rule-based study. None of the trained models outperformed the ruled-based algorithms for initial contact, four models for stair ascent FO and 13 models for stair descent FO outperformed the rule-based method in Chapter 4. These models were indicated by < symbol in the table below.

A few classifiers had only one true detection; therefore, the standard deviation will be 0. The best timing error was compared and selected between the classifiers with at least 0.9 F1-score. The timeliest (least absolute timing error) were: 1-1-23 for stair ascent IC with 7.561 ms, 1-1-10 for stair ascent FO with 16.5311 ms, 1-1-10 for stair descent IC with 25.6378 ms, and 1-1-10 for stair descent FO with 16.5 ms. The most consistent (least standard deviation) were: 1-1-23 for stair ascent IC with 12.127 ms, 1-1-10 for stair ascent FO with 23.7925 ms, 1-1-12 for stair descent IC with 34.6533 ms, and 1-1-10 for stair descent FO with 22.0436 ms. The most useful (least likely to have a detection outside 50 ms) were: 1-1-23 for stair ascent IC with 0.0002, 1-1-10 for stair ascent FO with 0.0357, 1-1-12 for stair descent IC with 0.1499, and 1-1-10 for stair descent FO with 0.0233.

When compared with the rule-based method in Chapter 4, there was a significant improvement. Among the classifiers that have an F1-score above 0.9, 12 classifiers outperformed in usefulness, 17 in timeliness and 19 in consistency for stair ascent IC, 44 in usefulness, 44 in timeliness but none in consistency for stair ascent FO. However, none of the classifiers was more timely or useful for stair descent. Only 18 classifiers had better consistency in stair descent IC, and 13 classifiers in stair descent FO. The symbol of %, ^, and & were indicating those classifiers that outperformed the rule-based method.

5.3.3 Discussion

Matlab k-fold cross-validation partition the data into k folds. For each fold, a model is trained with out-of-fold observations, the other k-1 folds. The performance of the trained model is access using the in-fold data. This training and testing are repeated until all folds are validated once. Then, the average test error is calculated over all validation. This is a time-consuming process because the model would be trained k times. This process is deployed to protect the model from overfitting since the model produces consistent performance across all folds, including both trained and untrained data. If the testing of any one-fold is unsatisfactory, MATLAB start the entire process again from different partitioning. The available classifier does not take the order of data into account, so they are time-independent training.

Suppose the model has a similar accuracy between the test data and the train data. In that case, we can be confident that the cross-validation has to prevent overfitting, and a good generalisation has been achieved. Appendix C compared the accuracy of each trained model in both training and testing data.

In our training, cubic SVM and quadratic SVM took the longest time to train. This is due to the lack of generalisation of one-fold training to another; therefore, the 6-fold cross-validation is not fulfilled. Despite the long training time, cubic SVM models generally reported a low accuracy.

Table 5.8: Supervised learning classifier performance for initial contact of stair ascent

Output	Cost	Model	TP	FP	FN	TPR	PPV	F1-score	mn(Te)	Sd(Te)	mn(|Te|)	Sd(|Te|)	Pr(|Te|>50))	
2	1	1	9	0	784	0.0113	1.0000	0.0224	-1.1111	48.3333	34.4444	31.6667	0.3010	%
		4	76	0	717	0.0958	1.0000	0.1749	73.8158	24.0537	73.8158	24.0537	0.8389	^
		5	554	41	239	0.6986	0.9311	0.7983	45.5596	39.5105	52.0578	30.4236	0.4630	^
		7	708	104	85	0.8928	0.8719	0.8822	-57.2740	41.2540	58.3475	39.7191	0.5746	^
		8	706	61	87	0.8903	0.9205	0.9051	-57.2238	41.7449	58.4419	40.0193	0.5738	^
		15	477	168	316	0.6015	0.7395	0.6634	-3.0818	45.7838	34.1929	30.5625	0.2759	%
		18	1	0	792	0.0013	1.0000	0.0025	-90.0000	0.0000	90.0000	0.0000	-	^
		19	3	0	790	0.0038	1.0000	0.0075	6.6667	23.0940	20.0000	0.0000	0.0374	%^
		20	29	0	764	0.0366	1.0000	0.0706	-5.8621	39.9569	30.0000	26.4575	0.2157	%^
		22	5	0	788	0.0063	1.0000	0.0125	0.0000	17.3205	12.0000	10.9545	0.0039	%^
		24	61	4	732	0.0769	0.9385	0.1422	1.4754	43.0440	31.3115	29.2959	0.2457	%^
		25	783	125	10	0.9874	0.8623	0.9206	-54.4189	39.8865	56.9221	36.2197	0.5485	^
2	2	1	71	3	722	0.0895	0.9595	0.1638	-7.0423	53.5962	45.3521	28.9251	0.3550	%
		2	4	0	789	0.0050	1.0000	0.0100	-52.5000	18.9297	52.5000	18.9297	0.5525	^
		4	312	27	481	0.3934	0.9204	0.5512	40.5449	44.7504	52.2115	30.2813	0.4379	
		5	731	323	62	0.9218	0.6935	0.7916	-6.1149	41.5482	31.0397	28.2651	0.2338	%^
		7	765	372	28	0.9647	0.6728	0.7927	-61.4771	48.8001	67.8824	39.3928	0.6041	
		8	757	249	36	0.9546	0.7525	0.8416	-65.6011	46.3603	68.7186	41.5929	0.6381	
		10	4	5	789	0.0050	0.4444	0.0100	-102.5000	53.1507	102.5000	53.1507	0.8404	
		12	149	8	644	0.1879	0.9490	0.3137	-16.5772	38.7956	32.4161	26.9045	0.2376	%^
		15	477	168	316	0.6015	0.7395	0.6634	-3.0818	45.7838	34.1929	30.5625	0.2759	%
		16	683	436	110	0.8613	0.6104	0.7144	-8.3309	42.6728	31.1127	30.3488	0.2502	%^
		17	169	11	624	0.2131	0.9389	0.3474	-9.4083	41.6148	31.7751	28.3754	0.2414	%^
		18	674	539	119	0.8499	0.5556	0.6720	-12.5964	41.4047	30.9347	30.2469	0.2485	%^
		19	684	445	109	0.8625	0.6058	0.7118	-7.6608	40.6077	29.6491	28.7646	0.2264	%^
		20	666	404	127	0.8398	0.6224	0.7150	-7.4625	41.9776	30.6456	29.6197	0.2410	%^
		22	77	5	716	0.0971	0.9390	0.1760	5.5844	43.9645	31.8182	30.6391	0.2592	%
		24	5	0	788	0.0063	1.0000	0.0125	-2.0000	44.3847	34.0000	23.0217	0.2604	%
		25	784	119	9	0.9887	0.8682	0.9245	-54.9872	40.4287	57.5893	36.6221	0.5538	^

1	1	1	785	295	8	0.9899	0.7269	0.8382	21.6927	20.1560	21.6927	20.1560	0.0803	%^
		2	785	272	8	0.9899	0.7427	0.8486	17.9880	19.6315	17.9880	19.6315	0.0517	%^
		3	784	67	9	0.9887	0.9213	0.9538	41.3622	38.0748	41.3622	38.0748	0.4185	%^
		4	785	166	8	0.9899	0.8254	0.9002	12.5877	15.1022	12.5877	15.1022	0.0066	%^%
		5	785	149	8	0.9899	0.8405	0.9091	18.1322	20.1497	18.1322	20.1497	0.0572	%^%
		6	785	106	8	0.9899	0.8810	0.9323	18.1461	18.7929	18.1461	18.7929	0.0452	%^%
		7	785	266	8	0.9899	0.7469	0.8514	24.6246	24.2028	24.6246	24.2028	0.1482	%^
		8	784	131	9	0.9887	0.8568	0.9180	32.2402	26.3491	32.2402	26.3491	0.2511	%^&
		9	785	96	8	0.9899	0.8910	0.9379	18.0220	18.4955	18.0220	18.4955	0.0420	%^&
		10	785	80	8	0.9899	0.9075	0.9469	20.4439	19.8703	20.4439	19.8703	0.0686	%^&
		11	730	1606	63	0.9206	0.3125	0.4666	41.1268	33.1288	41.1268	33.1288	0.3974	%^
		12	785	88	8	0.9899	0.8992	0.9424	22.4419	19.7853	22.4419	19.7853	0.0820	%^&
		13	785	72	8	0.9899	0.9160	0.9515	20.7604	19.4051	20.7604	19.4051	0.0661	%^&
		14	785	78	8	0.9899	0.9096	0.9481	18.4973	18.5999	18.4973	18.5999	0.0453	%^&
		15	785	896	8	0.9899	0.4670	0.6346	20.3812	20.2698	20.3812	20.2698	0.0722	%^
		16	785	283	8	0.9899	0.7350	0.8436	23.1749	21.5740	23.1749	21.5740	0.1072	%^
		17	784	140	9	0.9887	0.8485	0.9132	22.6512	20.4710	22.6512	20.4710	0.0910	%^&
		18	784	380	9	0.9887	0.6735	0.8012	22.5063	20.6505	22.5063	20.6505	0.0918	%^
		19	785	284	8	0.9899	0.7343	0.8432	23.3987	21.6512	23.3987	21.6512	0.1100	%^
		20	785	329	8	0.9899	0.7047	0.8233	21.8834	21.0946	21.8834	21.0946	0.0916	%^
		21	785	160	8	0.9899	0.8307	0.9033	21.5903	19.3098	21.5903	19.3098	0.0707	%^&
		22	785	340	8	0.9899	0.6978	0.8186	21.6520	20.4724	21.6520	20.4724	0.0833	%^
		23	782	49	11	0.9861	0.9410	0.9631	7.5610	12.1270	7.5610	12.1270	0.0002	%^&
		24	785	545	8	0.9899	0.5902	0.7395	19.3590	19.0480	19.3590	19.0480	0.0540	%^
		25	785	223	8	0.9899	0.7788	0.8717	22.5897	21.7301	22.5897	21.7301	0.1040	%^
3	1	1	708	146	85	0.8928	0.8290	0.8597	-7.6554	41.5422	31.4407	28.1874	0.2366	%^
		2	128	1	665	0.1614	0.9922	0.2777	-28.1250	35.1319	37.9688	24.0524	0.2798	%^
		4	661	113	132	0.8335	0.8540	0.8437	28.5628	44.4109	44.3570	28.6158	0.3531	%
		5	753	285	40	0.9496	0.7254	0.8225	-0.6773	38.7908	28.6454	26.1445	0.1975	%^
		7	768	485	25	0.9685	0.6129	0.7507	-41.6146	51.8145	55.6250	36.3401	0.4742	%
		8	763	401	30	0.9622	0.6555	0.7798	-41.7693	49.2105	51.4155	39.0079	0.4647	%

		12	601	60	192	0.7579	0.9092	0.8267	-5.2246	40.8410	31.7804	26.1469	0.2246	%^
		15	765	1049	28	0.9647	0.4217	0.5869	2.3791	36.9106	26.0131	26.2773	0.1764	%^
		16	672	279	121	0.8474	0.7066	0.7706	1.5179	43.4443	32.0833	29.3060	0.2501	%
		17	577	94	216	0.7276	0.8599	0.7883	-1.2652	41.2137	31.4905	26.5858	0.2253	%^
		18	684	352	109	0.8625	0.6602	0.7479	0.1754	43.8482	33.0117	28.8327	0.2542	%
		19	676	274	117	0.8525	0.7116	0.7757	1.3462	42.0700	31.4349	27.9658	0.2349	%^
		20	727	418	66	0.9168	0.6349	0.7503	0.2889	40.0007	29.6424	26.8376	0.2113	%^
		21	17	0	776	0.0214	1.0000	0.0420	7.6471	46.9746	40.5882	22.7680	0.2935	%
		22	684	354	109	0.8625	0.6590	0.7471	4.8246	40.8919	31.3158	26.7086	0.2246	%^
		23	119	3	674	0.1501	0.9754	0.2601	51.0084	35.6155	55.2101	28.6075	0.5136	^
		24	694	600	99	0.8752	0.5363	0.6651	4.6974	41.0168	30.1441	28.1868	0.2259	%^
		25	783	106	10	0.9874	0.8808	0.9310	-61.3410	39.9887	62.4904	38.1651	0.6143	^
3	2	1	785	426	8	0.9899	0.6482	0.7834	-18.6242	43.2115	34.6752	31.7906	0.2900	%^
		2	784	158	9	0.9887	0.8323	0.9037	-34.6556	45.5940	43.3036	37.4668	0.3999	%
		3	778	165	15	0.9811	0.8250	0.8963	-49.2159	41.5718	52.4807	37.3600	0.5010	^
		4	772	220	21	0.9735	0.7782	0.8650	-21.9301	44.5211	37.1632	32.8753	0.3173	%
		5	779	235	14	0.9823	0.7682	0.8622	-41.4249	42.1181	48.2542	34.0691	0.4343	^
		7	773	154	20	0.9748	0.8339	0.8988	-61.8629	47.6999	68.9521	36.6974	0.6077	
		8	774	240	19	0.9760	0.7633	0.8567	-51.6279	48.6553	59.5349	38.5647	0.5317	
		9	759	117	34	0.9571	0.8664	0.9095	-30.2372	41.4028	39.3808	32.8145	0.3429	%^
		10	779	363	14	0.9823	0.6821	0.8052	-30.6418	37.9466	38.5494	29.8682	0.3218	%^
		12	777	479	16	0.9798	0.6186	0.7584	-10.9395	36.7531	28.6229	25.5008	0.1926	%^
		13	786	307	7	0.9912	0.7191	0.8335	-25.0763	38.7289	34.0585	31.1142	0.2862	%^
		14	769	119	24	0.9697	0.8660	0.9149	-32.2367	37.3007	37.9064	31.5141	0.3307	%^
		15	765	1048	28	0.9647	0.4220	0.5871	2.3791	36.9106	26.0131	26.2773	0.1764	%^
		16	786	932	7	0.9912	0.4575	0.6260	-9.3384	35.9663	26.4631	26.0711	0.1786	%^
		17	785	564	8	0.9899	0.5819	0.7330	-14.9682	40.2800	30.6879	30.0643	0.2456	%^
		18	789	1387	4	0.9950	0.3626	0.5315	-12.3701	37.2750	28.1622	27.3591	0.2035	%^
		19	786	957	7	0.9912	0.4509	0.6199	-8.7277	35.6152	26.4631	25.3678	0.1728	%^
		20	786	960	7	0.9912	0.4502	0.6191	-8.1679	35.4689	25.8270	25.6312	0.1696	%^
		21	782	173	11	0.9861	0.8188	0.8947	-36.2660	41.9788	41.6368	36.6508	0.3917	%^

22	748	575	45	0.9433	0.5654	0.7070	-0.8957	36.8160	27.4733	24.5038	0.1746	%^
23	634	73	159	0.7995	0.8967	0.8453	-46.7666	42.6913	50.2681	38.4998	0.4815	^
24	536	267	257	0.6759	0.6675	0.6717	7.6119	47.2643	35.2985	32.3056	0.2963	%
25	783	102	10	0.9874	0.8847	0.9333	-60.7918	61.9413	39.9025	38.0914	0.6060	%

mn(Te) and sd(Te) represents the mean of timing error and the standard deviation of the timing error, respectively. Similarly, the next two columns are for the absolute timing error, |Te|. Pr(|Te|>50) is the likelihood for detection to have a fundamental timing error greater than 50 ms. Models which outperform the rule-based approach in chapter 4 are indicated with the symbol < for F1-score, % for the mean of absolute timing error, ^ for the standard deviation of timing error, and & for the likelihood to have a |T2|> 50 ms

Table 5.9: Supervised learning classifier performance for end contact of stair ascent

Output	Cost	Model	TP	FP	FN	TPR	PPV	F1-score	mn(Te)	Sd(Te)	mn(|Te|)	Sd(|Te|)	pr(|Te|>50)	
2	1	1	15	0	779	0.0189	1.0000	0.0371	0.6667	30.8143	22.0000	20.7709	0.1048	%
		4	46	0	748	0.0579	1.0000	0.1095	28.2609	12.3476	28.2609	12.3476	0.0392	%^
		5	730	49	64	0.9194	0.9371	0.9282	-6.4521	26.5354	19.3562	19.2520	0.0671	%&
		7	759	22	35	0.9559	0.9718	0.9638	-26.2582	27.0047	31.0540	21.3080	0.1920	%&
		8	768	69	26	0.9673	0.9176	0.9418	-28.3333	27.0416	33.1771	20.8067	0.2134	%&
		15	500	149	294	0.6297	0.7704	0.6930	3.6400	34.0333	23.4400	24.9200	0.1441	%
		16	79	2	715	0.0995	0.9753	0.1806	-2.6582	28.6763	21.3924	19.1307	0.0825	%
		18	57	1	737	0.0718	0.9828	0.1338	2.6316	27.6797	19.8246	19.3179	0.0721	%
		19	78	1	716	0.0982	0.9873	0.1787	-2.9487	28.6542	21.1538	19.4055	0.0826	%
		20	85	2	709	0.1071	0.9770	0.1930	4.8235	39.0242	25.2941	29.9837	0.2035	%
		22	69	1	725	0.0869	0.9857	0.1597	-4.4928	26.5428	20.7246	17.0051	0.0633	%
		24	179	10	615	0.2254	0.9471	0.3642	7.2626	32.9929	22.2346	25.3854	0.1389	%
		25	784	62	10	0.9874	0.9267	0.9561	-35.5357	26.3432	38.5969	21.6041	0.2921	%&
2	2	1	725	47	69	0.9131	0.9391	0.9259	-3.6138	24.5549	18.2069	16.8545	0.0439	%&
		4	381	8	413	0.4798	0.9794	0.6441	16.4042	26.7234	21.6010	22.7181	0.1108	%
		5	773	85	21	0.9736	0.9009	0.9358	-21.4360	27.3366	27.6973	20.9583	0.1525	%&
		7	775	35	19	0.9761	0.9568	0.9663	-34.8645	27.1697	38.0903	22.4176	0.2896	%&
		8	777	53	17	0.9786	0.9361	0.9569	-34.5946	27.7344	38.6873	21.6539	0.2904	%&
		10	766	42	28	0.9647	0.9480	0.9563	122.3107	34.2198	124.1123	26.9530	0.9827	
		12	441	15	353	0.5554	0.9671	0.7056	-8.1406	28.7100	20.6122	21.5604	0.0938	%
		15	500	149	294	0.6297	0.7704	0.6930	3.6400	34.0333	23.4400	24.9200	0.1441	%
		16	719	354	75	0.9055	0.6701	0.7702	-0.7371	30.6410	20.8762	22.4275	0.1028	%
		17	478	35	316	0.6020	0.9318	0.7314	-1.9665	30.6584	21.4644	21.9574	0.1036	%
		18	751	494	43	0.9458	0.6032	0.7366	-1.1718	25.5126	16.9640	19.0815	0.0503	%
		19	713	335	81	0.8980	0.6803	0.7742	-0.0842	31.6227	21.4306	23.2397	0.1138	%
		20	705	326	89	0.8879	0.6838	0.7726	-0.7943	30.6500	20.7092	22.5959	0.1029	%
		22	233	16	561	0.2935	0.9357	0.4468	3.6481	29.4343	21.4163	20.4721	0.0918	%
		23	6	0	788	0.0076	1.0000	0.0150	41.6667	19.4079	41.6667	19.4079	0.3338	%^
		24	47	1	747	0.0592	0.9792	0.1116	14.8936	33.1579	27.6596	23.3325	0.1700	%

		25	784	59	10	0.9874	0.9300	0.9578	-35.8291	26.4517	38.9413	21.6031	0.2967	%&
1	1	1	788	292	6	0.9924	0.7296	0.8410	-0.8249	25.7336	18.7183	17.6657	0.0521	%
		2	788	269	6	0.9924	0.7455	0.8514	2.8426	25.3673	18.2234	17.8627	0.0501	%
		3	787	65	7	0.9912	0.9237	0.9563	0.1779	25.7403	18.3990	17.9901	0.0521	%&
		4	783	169	11	0.9861	0.8225	0.8969	-4.5211	25.4272	18.4419	18.0685	0.0528	%
		5	787	148	7	0.9912	0.8417	0.9104	-4.6760	27.1081	19.2122	19.6765	0.0691	%&
		6	782	110	12	0.9849	0.8767	0.9276	-0.9079	25.8295	17.5320	18.9796	0.0530	%&
		7	787	265	7	0.9912	0.7481	0.8527	1.2579	25.2230	17.9288	17.7745	0.0477	%
		8	786	130	8	0.9899	0.8581	0.9193	11.8193	28.5108	21.2341	22.3892	0.1053	%&
		9	783	99	11	0.9861	0.8878	0.9344	-2.2095	26.8194	18.0204	19.9754	0.0632	%&
		10	787	79	7	0.9912	0.9088	0.9482	-0.6734	23.7925	16.5311	17.1147	0.0357	%&
		11	533	1803	261	0.6713	0.2282	0.3406	49.7186	38.3165	54.9343	30.3526	0.5017	
		12	785	87	9	0.9887	0.9002	0.9424	4.9299	26.3421	19.1720	18.7138	0.0621	%&
		13	787	71	7	0.9912	0.9172	0.9528	3.7103	24.7575	17.5349	17.8564	0.0458	%&
		14	787	77	7	0.9912	0.9109	0.9493	-0.7624	24.7774	17.2554	17.7870	0.0437	%&
		15	791	891	3	0.9962	0.4703	0.6389	3.1226	31.2561	20.9987	23.3496	0.1114	%
		16	788	281	6	0.9924	0.7371	0.8459	3.4898	28.7985	20.3173	20.6938	0.0848	%
		17	787	138	7	0.9912	0.8508	0.9156	4.9047	26.2128	19.0343	18.6663	0.0608	%&
		18	787	378	7	0.9912	0.6755	0.8035	0.1271	26.2932	19.0597	18.1001	0.0572	%
		19	788	282	6	0.9924	0.7364	0.8455	3.6421	28.8854	20.6218	20.5390	0.0859	%
		20	789	325	5	0.9937	0.7083	0.8270	5.5260	30.5139	21.4195	22.4120	0.1069	%
		21	788	158	6	0.9924	0.8330	0.9057	4.0355	26.6410	19.0355	19.0588	0.0635	%&
		22	787	339	7	0.9912	0.6989	0.8198	3.3926	27.1889	19.5299	19.2058	0.0680	%
		23	782	49	12	0.9849	0.9410	0.9625	4.6803	26.0225	17.9028	19.4469	0.0586	%&
		24	787	544	7	0.9912	0.5913	0.7407	3.8755	25.0719	17.6239	18.2385	0.0487	%
		25	788	220	6	0.9924	0.7817	0.8746	1.0152	25.5060	18.1726	17.9144	0.0501	%
3	1	1	773	118	21	0.9736	0.8676	0.9175	12.9495	25.1955	20.4528	19.5924	0.0769	%&
		4	711	106	83	0.8955	0.8703	0.8827	10.3797	26.6008	20.9845	19.3526	0.0798	%
		5	779	139	15	0.9811	0.8486	0.9100	-12.4904	25.7826	21.0655	19.4069	0.0805	%&
		7	779	12	15	0.9811	0.9848	0.9830	-20.4236	26.5826	26.7137	20.2423	0.1370	%&
		8	784	8	10	0.9874	0.9899	0.9887	-21.6837	28.3424	28.7755	21.0947	0.1646	<%&

		9	566	13	228	0.7128	0.9775	0.8245	9.7703	29.0215	20.8657	22.3993	0.1026	%
		10	759	36	35	0.9559	0.9547	0.9553	-1.3439	25.8683	18.9460	17.6508	0.0536	%&
		12	747	116	47	0.9408	0.8656	0.9016	6.2517	28.6400	20.6827	20.7614	0.0881	%&
		13	752	23	42	0.9471	0.9703	0.9586	0.8511	26.4514	19.3351	18.0570	0.0589	%&
		14	394	5	400	0.4962	0.9875	0.6605	14.5939	27.0888	21.0914	22.3909	0.1042	%
		15	782	702	12	0.9849	0.5270	0.6866	6.5985	28.5894	19.8465	21.5999	0.0884	%
		16	776	276	18	0.9773	0.7376	0.8407	7.3840	29.8597	21.4562	22.0280	0.1041	%
		17	749	78	45	0.9433	0.9057	0.9241	11.3351	29.3482	22.3097	22.1717	0.1122	%&
		18	775	301	19	0.9761	0.7203	0.8289	2.2581	27.9153	19.7032	19.8909	0.0742	%
		19	774	286	20	0.9748	0.7302	0.8350	7.8553	29.4854	21.5504	21.5907	0.1013	%
		20	764	294	30	0.9622	0.7221	0.8251	9.0969	29.0631	21.4529	21.6031	0.1007	%
		21	684	13	110	0.8615	0.9813	0.9175	11.6959	25.9922	20.3216	19.9754	0.0791	%&
		22	769	332	25	0.9685	0.6985	0.8116	5.7737	27.1321	19.6099	19.6082	0.0715	%
		23	512	6	282	0.6448	0.9884	0.7805	20.3516	27.9969	25.2734	23.6396	0.1508	%
		24	767	644	27	0.9660	0.5436	0.6957	9.5437	26.0483	19.6089	19.6139	0.0713	%
		25	786	23	8	0.9899	0.9716	0.9807	-24.7201	25.8212	28.8931	21.0410	0.1657	%&
3	2	1	787	274	7	0.9912	0.7418	0.8485	-12.7192	26.0109	21.3088	19.5936	0.0838	%
		2	785	10	9	0.9887	0.9874	0.9880	-12.9427	24.9618	20.5350	19.1987	0.0747	<%&
		3	784	19	10	0.9874	0.9763	0.9818	-24.8724	28.3381	29.8980	22.9662	0.1917	%&
		4	785	278	9	0.9887	0.7385	0.8454	-7.1847	27.5736	20.9427	19.3089	0.0793	%
		5	781	137	13	0.9836	0.8508	0.9124	-23.9052	26.4572	29.0781	20.6291	0.1646	%&
		7	779	12	15	0.9811	0.9848	0.9830	-29.2811	25.6784	32.8241	20.9537	0.2109	%&
		8	785	18	9	0.9887	0.9776	0.9831	-29.2357	27.6536	34.2038	21.1930	0.2284	%&
		9	781	94	13	0.9836	0.8926	0.9359	-12.2151	28.9488	23.1498	21.2329	0.1117	%&
		10	100	290	694	0.1259	0.2564	0.1689	11.2000	49.3448	35.6000	35.7974	0.3233	%
		12	776	114	18	0.9773	0.8719	0.9216	-9.0464	27.4953	21.1082	19.7940	0.0841	%&
		13	786	38	8	0.9899	0.9539	0.9716	-13.4860	26.7722	22.0102	20.3418	0.0952	%&
		14	786	15	8	0.9899	0.9813	0.9856	-16.5013	26.8204	23.5242	20.9257	0.1124	<%&
		15	782	702	12	0.9849	0.5270	0.6866	6.5985	28.5894	19.8465	21.5999	0.0884	%
		16	788	227	6	0.9924	0.7764	0.8712	-7.7411	27.8629	20.6091	20.2747	0.0838	%
		17	787	45	7	0.9912	0.9459	0.9680	-10.0381	26.2862	20.3558	19.4158	0.0754	%&

18	789	270	5	0.9937	0.7450	0.8516	-11.9392	26.2247	21.0139	19.7056	0.0824	%
19	786	227	8	0.9899	0.7759	0.8700	-7.5318	27.1399	20.2290	19.5868	0.0758	%
20	787	268	7	0.9912	0.7460	0.8513	-6.1245	28.0240	20.2287	20.3268	0.0813	%
21	786	10	8	0.9899	0.9874	0.9887	-14.2239	24.9324	21.1196	19.4319	0.0807	<%&
22	777	379	17	0.9786	0.6721	0.7969	1.1197	25.7847	18.4170	18.0689	0.0527	%
23	757	67	37	0.9534	0.9187	0.9357	2.9855	28.1637	20.6341	19.3852	0.0775	%&
24	748	620	46	0.9421	0.5468	0.6920	11.5107	26.6815	19.4519	21.5800	0.0851	%
25	786	22	8	0.9899	0.9728	0.9813	-28.7913	25.7561	32.3028	21.1796	0.2062	%&

mn(Te) and sd(Te) represents the mean of timing error and the standard deviation of the timing error, respectively. Similarly, the next two columns are for the absolute timing error, |Te|. Pr(|Te|>50) is the likelihood for detection to have a fundamental timing error greater than 50 ms. Models which outperform the rule-based approach in chapter 4 are indicated with the symbol < for F1-score, % for the mean of absolute timing error, ^ for the standard deviation of timing error, and & for the likelihood to have a |T2|> 50 ms

Table 5.10: Supervised learning classifier performance for initial contact of stair descent

Output	Cost	Model	TP	FP	FN	TPR	PPV	F1-score	mn(Te)	Sd(Te)	mn(|Te|)	Sd(|Te|)	pr(|Te|>50)	
2	1	1	34	0	845	0.0387	1.0000	0.0745	-14.4118	51.7682	40.8824	34.2334	0.3526	
		5	192	29	687	0.2184	0.8688	0.3491	-15.8333	36.7863	29.4792	27.0493	0.2133	%
		7	777	51	102	0.8840	0.9384	0.9104	-23.4749	49.7030	42.0592	35.3682	0.3665	
		8	851	31	28	0.9681	0.9649	0.9665	-28.8954	50.3419	44.5711	37.1661	0.3961	
		15	477	163	402	0.5427	0.7453	0.6280	-8.3438	43.1869	32.0335	30.1093	0.2557	%
		16	1	0	878	0.0011	1.0000	0.0023	-10.0000	0.0000	10.0000	0.0000	-	%^
		18	5	0	874	0.0057	1.0000	0.0113	6.0000	32.8634	22.0000	22.8035	0.1345	%
		19	1	0	878	0.0011	1.0000	0.0023	-10.0000	0.0000	10.0000	0.0000	-	%^
		20	48	2	831	0.0546	0.9600	0.1033	-21.4583	47.8913	37.7083	36.2168	0.3434	
		22	11	0	868	0.0125	1.0000	0.0247	-16.3636	39.3123	32.7273	25.7258	0.2418	%
		24	116	8	763	0.1320	0.9355	0.2313	-13.4483	45.1491	34.1379	32.3317	0.2891	%
		25	876	215	3	0.9966	0.8029	0.8893	-36.6438	53.3638	50.4795	40.5082	0.4534	
2	2	1	157	9	722	0.1786	0.9458	0.3005	-13.8854	52.4972	37.7070	38.9752	0.3576	
		4	36	4	843	0.0410	0.9000	0.0783	59.7222	75.1184	86.3889	40.4371	0.6235	
		5	766	167	113	0.8714	0.8210	0.8455	-27.9373	40.9741	36.5535	33.5035	0.3237	%
		7	861	177	18	0.9795	0.8295	0.8983	-28.3972	52.1490	46.0046	37.5230	0.4057	
		8	875	61	4	0.9954	0.9348	0.9642	-33.9086	51.6261	47.2114	39.8107	0.4297	
		10	616	290	263	0.7008	0.6799	0.6902	35.9253	78.1459	66.6396	54.3273	0.5643	
		11	5	1	874	0.0057	0.8333	0.0113	20.0000	14.1421	20.0000	14.1421	0.0169	%^
		12	280	17	599	0.3185	0.9428	0.4762	-16.6786	41.9255	32.2500	31.5139	0.2692	%
		15	477	163	402	0.5427	0.7453	0.6280	-8.3438	43.1869	32.0335	30.1093	0.2557	%
		16	677	383	202	0.7702	0.6387	0.6983	-9.2319	39.2835	26.7208	30.2240	0.2155	%
		17	241	35	638	0.2742	0.8732	0.4173	-14.3154	34.3943	27.3444	25.2570	0.1805	%
		18	679	501	200	0.7725	0.5754	0.6595	-6.1267	38.7396	28.0412	27.4021	0.2024	%
		19	669	365	210	0.7611	0.6470	0.6994	-9.5366	39.5589	26.9357	30.4855	0.2193	%
		20	656	384	223	0.7463	0.6308	0.6837	-7.2104	39.0458	26.4787	29.5711	0.2080	%
		22	97	10	782	0.1104	0.9065	0.1968	-14.6392	38.3260	30.1031	27.7449	0.2239	%
		24	17	0	862	0.0193	1.0000	0.0379	-8.2353	46.2649	29.4118	35.9636	0.2874	%
		25	877	246	2	0.9977	0.7809	0.8761	-33.0901	52.5092	47.7765	39.6008	0.4305	

1	1	1	879	559	0	1.0000	0.6113	0.7587	1.7975	37.8307	28.6234	24.7823	0.1868	%
		2	879	53	0	1.0000	0.9431	0.9707	16.0637	39.2263	34.4937	24.6143	0.2396	%
		3	879	56	0	1.0000	0.9401	0.9691	14.6530	41.2348	35.0853	26.1320	0.2541	%
		4	877	66	2	0.9977	0.9300	0.9627	-23.1129	52.9194	44.5040	36.7751	0.3892	
		5	879	444	0	1.0000	0.6644	0.7984	19.6246	48.2894	43.1513	29.2108	0.3393	
		6	879	30	0	1.0000	0.9670	0.9832	-2.7190	49.5883	37.9863	31.9655	0.3140	
		7	879	60	0	1.0000	0.9361	0.9670	9.4994	42.4288	34.3003	26.6963	0.2503	%
		8	879	85	0	1.0000	0.9118	0.9539	15.2787	41.7704	36.3026	25.6728	0.2620	%
		9	879	13	0	1.0000	0.9854	0.9927	-7.3606	48.9162	37.5768	32.1468	0.3122	
		10	878	38	1	0.9989	0.9585	0.9783	-0.6492	34.9087	25.6378	23.6853	0.1521	%
		12	879	123	0	1.0000	0.8772	0.9346	2.1729	34.6533	26.4050	22.5291	0.1499	%
		13	879	32	0	1.0000	0.9649	0.9821	6.8373	37.0527	29.5449	23.3625	0.1845	%
		14	879	10	0	1.0000	0.9888	0.9943	10.1251	39.6839	32.4460	24.9702	0.2224	%
		15	879	1402	0	1.0000	0.3854	0.5563	4.3231	32.3064	23.3902	22.6868	0.1250	%
		16	879	456	0	1.0000	0.6584	0.7940	10.0000	35.0528	28.5666	22.6238	0.1704	%
		17	879	160	0	1.0000	0.8460	0.9166	11.4448	37.4145	31.4448	23.2610	0.2017	%
		18	879	558	0	1.0000	0.6117	0.7591	10.3413	32.5264	26.8828	21.0125	0.1432	%
		19	879	462	0	1.0000	0.6555	0.7919	9.6815	35.1390	28.4300	22.7904	0.1703	%
		20	879	544	0	1.0000	0.6177	0.7637	7.0193	35.4405	27.4289	23.4979	0.1664	%
		21	879	48	0	1.0000	0.9482	0.9734	10.5688	38.5606	32.3663	23.4515	0.2114	%
		22	879	479	0	1.0000	0.6473	0.7859	5.0398	33.1624	25.8589	21.3477	0.1361	%
		23	877	15	2	0.9977	0.9832	0.9904	-29.7605	51.1628	45.1767	38.2239	0.4057	
		24	879	1001	0	1.0000	0.4676	0.6372	8.6462	32.8261	26.0523	21.7461	0.1409	%
>		25	879	33	0	1.0000	0.9638	0.9816	16.3481	39.0254	34.4141	24.5940	0.2388	%
3	1	1	754	114	125	0.8578	0.8687	0.8632	17.8515	34.6049	31.8833	22.3317	0.2014	%
		4	117	20	762	0.1331	0.8540	0.2303	57.0940	53.8046	68.0342	38.9128	0.5757	
		5	812	94	67	0.9238	0.8962	0.9098	-7.5739	39.8320	30.6773	26.4908	0.2176	%
		7	862	184	17	0.9807	0.8241	0.8956	-11.8794	48.0243	37.8422	31.8421	0.3124	
		8	873	73	6	0.9932	0.9228	0.9567	-22.9668	49.3840	41.4777	35.2773	0.3618	
		10	704	56	175	0.8009	0.9263	0.8591	6.8182	27.5059	21.9602	17.8939	0.0776	%^
		11	861	124	18	0.9795	0.8741	0.9238	-30.2787	46.1108	40.1742	37.7919	0.3753	

		12	805	107	74	0.9158	0.8827	0.8989	6.3851	34.7876	27.0807	22.7314	0.1575	%
		13	577	26	302	0.6564	0.9569	0.7787	13.4142	25.2754	23.6742	16.0517	0.0799	%^
		15	862	895	17	0.9807	0.4906	0.6540	5.8701	30.9076	22.3898	22.0883	0.1120	%
		16	806	299	73	0.9170	0.7294	0.8125	10.8313	35.2197	28.7717	23.0011	0.1751	%
		17	719	114	160	0.8180	0.8631	0.8400	17.3018	34.0442	31.0153	22.2595	0.1924	%
		18	771	342	108	0.8771	0.6927	0.7741	12.5681	33.0783	28.1842	21.3766	0.1582	%
		19	819	296	60	0.9317	0.7345	0.8215	10.7448	36.0529	29.3529	23.5108	0.1841	%
		20	841	387	38	0.9568	0.6849	0.7983	8.3353	35.5446	27.7170	23.7451	0.1709	%
		22	825	257	54	0.9386	0.7625	0.8414	7.5152	35.8192	28.0970	23.4343	0.1720	%
		24	659	268	220	0.7497	0.7109	0.7298	16.4795	35.1556	31.3505	22.8813	0.1995	%
		25	875	238	4	0.9954	0.7862	0.8785	-22.6629	48.5098	40.7200	34.7473	0.3536	
3	2	1	861	185	18	0.9795	0.8231	0.8945	-16.0627	44.4390	35.3659	31.3200	0.2911	%
		2	842	206	37	0.9579	0.8034	0.8739	-10.6295	43.7967	34.4537	29.0311	0.2675	%
		3	806	60	73	0.9170	0.9307	0.9238	-16.2655	44.0590	35.6948	30.5022	0.2882	%
		4	733	448	146	0.8339	0.6207	0.7117	-8.9768	52.1129	39.5362	35.0885	0.3445	
		5	872	112	7	0.9920	0.8862	0.9361	-31.0436	43.6449	40.2408	35.3342	0.3637	
		7	876	319	3	0.9966	0.7331	0.8447	-18.5845	49.3344	40.4566	33.7794	0.3444	
		8	875	180	4	0.9954	0.8294	0.9049	-25.9429	51.0137	43.8857	36.7157	0.3869	
		9	859	492	20	0.9772	0.6358	0.7704	-26.3562	49.8984	44.4005	34.8083	0.3808	
		10	877	87	2	0.9977	0.9098	0.9517	-20.2166	39.3651	31.7788	30.7855	0.2619	%
		12	877	183	2	0.9977	0.8274	0.9046	-15.2452	42.3520	31.7788	31.8642	0.2676	%
		13	877	47	2	0.9977	0.9491	0.9728	-19.6123	40.2627	32.1779	31.1379	0.2671	%
		14	866	97	13	0.9852	0.8993	0.9403	-20.2887	41.7788	34.7691	30.7780	0.2847	%
		15	862	895	17	0.9807	0.4906	0.6540	5.8701	30.9076	22.3898	22.0883	0.1120	%
		16	878	596	1	0.9989	0.5957	0.7463	-7.6310	38.3121	28.0410	27.1830	0.2006	%
		17	879	283	0	1.0000	0.7565	0.8613	-13.7201	45.1792	34.9943	31.6799	0.2902	%
		18	879	1028	0	1.0000	0.4609	0.6310	-10.1138	39.0226	29.4084	27.5560	0.2151	%
		19	878	589	1	0.9989	0.5985	0.7485	-8.8610	39.3948	28.5877	28.5021	0.2158	%
		20	879	651	0	1.0000	0.5745	0.7298	-6.5757	37.6894	27.1900	26.9006	0.1913	%
		21	863	120	16	0.9818	0.8779	0.9270	-12.9085	43.6714	33.7196	30.5890	0.2727	%
		22	863	363	16	0.9818	0.7039	0.8200	2.6883	35.5299	26.4890	23.8144	0.1605	%

23	519	131	360	0.5904	0.7985	0.6789	-9.4412	56.8093	44.4316	36.5872	0.3853	
24	832	620	47	0.9465	0.5730	0.7139	11.5264	31.7322	25.9736	21.5526	0.1389	%
25	876	247	3	0.9966	0.7801	0.8751	-22.2489	48.4181	40.4452	34.6728	0.3511	

mn(Te) and sd(Te) represents the mean of timing error and the standard deviation of the timing error, respectively. Similarly, the next two columns are for the absolute timing error, |Te|. Pr(|Te|>50) is the likelihood for detection to have a fundamental timing error greater than 50 ms. Models which outperform the rule-based approach in chapter 4 are indicated with the symbol < for F1-score, % for the mean of absolute timing error, ^ for the standard deviation of timing error, and & for the likelihood to have a |T2|> 50 ms

Table 5.11: Supervised learning classifier performance for end contact of stair descent

Output	Cost	Model	TP	FP	FN	TPR	PPV	F1-score	mn(Te)	Sd(Te)	mn(|Te|)	Sd(|Te|)	pr(|Te|>50)	
2	1	1	50	0	830	0.0568	1.0000	0.1075	12.2000	27.9424	24.6000	17.7523	0.1011	
		4	1	1	879	0.0011	0.5000	0.0023	0.0000	0.0000	0.0000	0.0000	-	^
		5	828	112	52	0.9409	0.8809	0.9099	-37.5362	23.3312	38.5990	21.5251	0.2967	
		7	872	2	8	0.9909	0.9977	0.9943	-54.8165	23.9635	55.0000	23.5388	0.5797	
		8	876	0	4	0.9955	1.0000	0.9977	-58.9269	25.3071	59.1324	24.8226	0.6379	<
		11	608	46	272	0.6909	0.9297	0.7927	13.9803	45.7766	36.0855	31.4159	0.2968	
		15	620	250	260	0.7045	0.7126	0.7086	-1.1774	30.6484	23.7258	19.4136	0.1031	
		16	91	2	789	0.1034	0.9785	0.1871	10.8791	23.2211	19.6703	16.3606	0.0504	%
		18	75	1	805	0.0852	0.9868	0.1569	18.2667	24.1825	24.6667	17.5016	0.0971	
		19	81	2	799	0.0920	0.9759	0.1682	10.4938	24.2333	19.6296	17.5673	0.0578	%
		20	104	4	776	0.1182	0.9630	0.2105	7.4038	29.1629	23.7500	18.3381	0.0966	
		22	91	4	789	0.1034	0.9579	0.1867	7.3626	28.7454	21.6484	20.1805	0.0920	%
		24	289	35	591	0.3284	0.8920	0.4801	-0.3114	34.2920	26.5398	21.6618	0.1448	
		25	880	0	0	1.0000	1.0000	1.0000	-70.2159	26.1147	70.2386	26.0535	0.7806	<
2	2	1	413	58	467	0.4693	0.8769	0.6114	5.0847	26.2513	20.8232	16.7448	0.0615	%
		2	228	1	652	0.2591	0.9956	0.4112	12.6754	26.7035	23.3772	18.0430	0.0906	
		4	449	44	431	0.5102	0.9108	0.6540	-27.5724	18.5558	28.4187	17.2286	0.1134	
		5	876	42	4	0.9955	0.9542	0.9744	-54.8858	26.5134	55.8447	24.4266	0.5731	
		7	880	0	0	1.0000	1.0000	1.0000	-62.7386	23.4484	62.8068	23.2650	0.7065	<
		8	880	0	0	1.0000	1.0000	1.0000	-66.6136	24.5623	66.6818	24.3764	0.7506	<
		10	646	1677	234	0.7341	0.2781	0.4034	40.8824	25.2122	42.3684	22.6218	0.3590	
		11	797	1539	83	0.9057	0.3412	0.4956	49.7742	25.0968	50.6274	23.3255	0.4964	
		12	540	24	340	0.6136	0.9574	0.7479	2.8889	24.3007	19.4815	14.7870	0.0410	%
		13	19	0	861	0.0216	1.0000	0.0423	20.5263	8.4811	20.5263	8.4811	0.0003	%
		15	620	250	260	0.7045	0.7126	0.7086	-1.1774	30.6484	23.7258	19.4136	0.1031	^
		16	815	546	65	0.9261	0.5988	0.7274	-3.3620	23.4282	17.6687	15.7363	0.0346	%
		17	591	17	289	0.6716	0.9720	0.7944	1.6920	23.6825	18.4772	14.8911	0.0352	%
		18	814	505	66	0.9250	0.6171	0.7403	1.5111	23.0738	16.9902	15.6737	0.0306	%
		19	820	552	60	0.9318	0.5977	0.7282	-2.5610	22.9886	17.3902	15.2398	0.0306	%

		20	805	558	75	0.9148	0.5906	0.7178	-2.6460	23.4272	17.8509	15.3881	0.0339	%
		21	210	1	670	0.2386	0.9953	0.3850	12.8095	23.9060	21.2857	16.7665	0.0642	%
		22	250	22	630	0.2841	0.9191	0.4340	2.6800	32.6417	25.0000	21.0993	0.1268	
		23	66	0	814	0.0750	1.0000	0.1395	-8.4848	16.0041	14.8485	10.2646	0.0049	%
		24	87	4	793	0.0989	0.9560	0.1792	4.3678	27.5217	22.5287	16.2265	0.0728	
		25	880	1	0	1.0000	0.9989	0.9994	-67.9773	25.9868	68.0227	25.8674	0.7555	<
1	1	1	880	558	0	1.0000	0.6120	0.7593	3.0568	22.5526	17.4205	14.6341	0.0280	%
		2	880	52	0	1.0000	0.9442	0.9713	5.4091	26.4694	21.7955	15.9480	0.0642	%
		3	880	55	0	1.0000	0.9412	0.9697	-5.8977	25.5695	20.3750	16.5230	0.0567	%
		4	880	64	0	1.0000	0.9322	0.9649	49.8636	37.8997	56.5455	26.9179	0.5028	
		5	880	443	0	1.0000	0.6652	0.7989	13.8068	22.2394	20.5568	16.1980	0.0539	%
		6	880	29	0	1.0000	0.9681	0.9838	30.8409	35.3590	40.2955	24.0194	0.3051	
		7	880	59	0	1.0000	0.9372	0.9676	-36.9432	24.3663	38.4205	21.9604	0.2962	
		8	880	84	0	1.0000	0.9129	0.9544	-42.6477	28.5670	43.7841	26.7908	0.3990	
		9	880	12	0	1.0000	0.9865	0.9932	36.0000	30.1555	40.5682	23.6469	0.3234	
		10	880	36	0	1.0000	0.9607	0.9800	-0.2955	22.0436	16.5000	14.6098	0.0233	%
		12	880	122	0	1.0000	0.8782	0.9352	3.3409	24.7577	19.2500	15.9102	0.0453	%
		13	880	31	0	1.0000	0.9660	0.9827	1.5795	22.7592	17.6250	14.4736	0.0284	%
		14	880	9	0	1.0000	0.9899	0.9949	6.0568	24.3649	19.7614	15.4732	0.0464	%
		15	880	1401	0	1.0000	0.3858	0.5568	0.7955	21.6720	16.5227	14.0355	0.0211	%
		16	880	455	0	1.0000	0.6592	0.7946	1.1364	26.3274	20.2727	16.8218	0.0578	%
		17	880	159	0	1.0000	0.8470	0.9171	2.5341	25.5196	20.0114	16.0240	0.0512	%
		18	880	557	0	1.0000	0.6124	0.7596	6.9318	21.1317	17.5000	13.7133	0.0243	%
		19	880	461	0	1.0000	0.6562	0.7924	1.1023	26.0705	20.0568	16.6779	0.0553	%
		20	880	543	0	1.0000	0.6184	0.7642	4.1477	25.1539	20.0341	15.7519	0.0498	%
		21	880	47	0	1.0000	0.9493	0.9740	3.7955	25.4990	20.5227	15.5871	0.0524	%
		22	880	478	0	1.0000	0.6480	0.7864	0.6364	26.1840	19.9545	16.9520	0.0563	%
		23	880	13	0	1.0000	0.9854	0.9927	53.5795	31.2941	56.6705	25.2619	0.5460	
		24	880	1000	0	1.0000	0.4681	0.6377	-0.1364	24.9227	18.9773	16.1435	0.0448	%
		25	880	32	0	1.0000	0.9649	0.9821	0.3636	25.5293	19.6818	16.2500	0.0502	%
3	1	1	867	202	13	0.9852	0.8110	0.8897	2.6874	25.4302	20.1038	15.7891	0.0505	%

		2	827	110	53	0.9398	0.8826	0.9103	-8.1378	29.6046	24.0750	19.0376	0.1035	
		4	492	16	388	0.5591	0.9685	0.7089	-17.4797	25.9513	26.1382	17.1766	0.1097	
		5	877	28	3	0.9966	0.9691	0.9826	-48.6203	23.5209	49.3729	21.8952	0.4766	
		7	880	0	0	1.0000	1.0000	1.0000	-56.4318	24.2333	56.5909	23.8590	0.6047	<
		8	880	0	0	1.0000	1.0000	1.0000	-60.3864	25.8478	60.5455	25.4725	0.6561	<
		9	237	1	643	0.2693	0.9958	0.4240	15.1899	23.3177	23.8819	14.2360	0.0703	
		10	735	35	145	0.8352	0.9545	0.8909	6.1088	38.8820	26.7619	28.8445	0.2040	
		11	866	703	14	0.9841	0.5519	0.7072	-68.9838	54.9735	82.2171	31.9201	0.6503	
		12	835	42	45	0.9489	0.9521	0.9505	4.3832	24.8968	19.6407	15.9018	0.0479	%
		13	850	1	30	0.9659	0.9988	0.9821	-1.0118	25.0025	19.3882	15.8051	0.0457	%
		14	555	0	325	0.6307	1.0000	0.7735	3.1892	24.0883	19.1532	14.9308	0.0396	%
		15	874	916	6	0.9932	0.4883	0.6547	2.0366	21.1517	16.3616	13.5477	0.0186	%
		16	858	209	22	0.9750	0.8041	0.8814	8.5897	24.1831	20.2914	15.6995	0.0511	%
		17	860	16	20	0.9773	0.9817	0.9795	4.0698	24.5508	19.4884	15.4624	0.0445	%
		18	850	225	30	0.9659	0.7907	0.8696	13.2941	21.3606	19.6941	15.6493	0.0444	%
		19	855	210	25	0.9716	0.8028	0.8792	8.5848	23.8503	20.0000	15.5613	0.0483	%
		20	865	312	15	0.9830	0.7349	0.8410	6.1387	23.8073	19.3179	15.1954	0.0419	%
		21	829	17	51	0.9420	0.9799	0.9606	-0.3378	26.4051	20.9650	16.0400	0.0583	%
		22	861	341	19	0.9784	0.7163	0.8271	2.8223	24.7838	19.3612	15.7136	0.0450	%
		23	88	0	792	0.1000	1.0000	0.1818	2.5000	16.3475	12.5000	10.7479	0.0025	%
		24	863	559	17	0.9807	0.6069	0.7498	5.4577	23.1607	18.4357	15.0324	0.0356	%
		25	880	11	0	1.0000	0.9877	0.9938	-58.5909	26.9042	59.0909	25.7862	0.6253	
3	2	1	878	256	2	0.9977	0.7743	0.8719	-28.8838	31.7542	34.2369	25.8854	0.2595	
		2	875	96	5	0.9943	0.9011	0.9454	-32.7314	31.4372	36.3657	27.1457	0.2956	
		3	880	0	0	1.0000	1.0000	1.0000	-23.3977	25.5172	28.3750	19.8279	0.1506	<
		4	861	42	19	0.9784	0.9535	0.9658	-56.5854	25.3799	57.5842	23.0211	0.6024	
		5	880	15	0	1.0000	0.9832	0.9915	-60.4659	24.4323	60.9205	23.2745	0.6658	
		7	880	0	0	1.0000	1.0000	1.0000	-64.4432	23.8484	64.4659	23.7868	0.7276	<
		8	880	0	0	1.0000	1.0000	1.0000	-67.6705	25.7609	67.7159	25.6410	0.7536	<
		9	866	0	14	0.9841	1.0000	0.9920	-48.7991	28.9538	49.8614	27.0812	0.4838	
		10	856	77	24	0.9727	0.9175	0.9443	-30.6192	39.0976	34.6612	35.5591	0.3297	

12	872	99	8	0.9909	0.8980	0.9422	-22.5115	28.7872	28.2225	23.2084	0.1757	
13	879	6	1	0.9989	0.9932	0.9960	-28.8623	28.3619	32.1388	24.5826	0.2308	<
14	878	0	2	0.9977	1.0000	0.9989	-42.1298	29.8861	43.3599	28.0696	0.3972	<
15	874	916	6	0.9932	0.4883	0.6547	2.0366	21.1517	16.3616	13.5477	0.0186	%
16	880	510	0	1.0000	0.6331	0.7753	-21.0455	28.4455	26.5227	23.4159	0.1606	
17	880	107	0	1.0000	0.8916	0.9427	-29.3523	26.9650	32.4886	23.0854	0.2235	
18	878	540	2	0.9977	0.6192	0.7641	-13.0979	27.0641	21.8907	20.6027	0.0962	
19	880	533	0	1.0000	0.6228	0.7676	-21.1023	28.0587	26.3523	23.1921	0.1572	
20	880	534	0	1.0000	0.6223	0.7672	-17.3523	27.9958	24.1477	22.3927	0.1298	
21	879	46	1	0.9989	0.9503	0.9740	-36.2457	30.6314	38.8396	27.2635	0.3291	
22	868	366	12	0.9864	0.7034	0.8212	-3.9631	27.2111	20.5300	18.2813	0.0690	%
23	874	3	6	0.9932	0.9966	0.9949	-50.5378	23.7891	51.3616	21.9517	0.5090	
24	869	693	11	0.9875	0.5563	0.7117	2.3936	23.5066	17.9287	15.3782	0.0343	%
25	880	7	0	1.0000	0.9921	0.9960	-57.9886	27.2339	58.6250	25.8339	0.6154	<

mn(Te) and sd(Te) represents the mean of timing error and the standard deviation of the timing error, respectively. Similarly, the next two columns are for the absolute timing error, |Te|. Pr(|Te|>50) is the likelihood for detection to have a fundamental timing error greater than 50 ms. Models which outperform the rule-based approach in chapter 4 are indicated with the symbol < for F1-score, % for the mean of absolute timing error, ^ for the standard deviation of timing error, and & for the likelihood to have a |T2|> 50 ms

The only improvement a transition output give is the reduction in the standard deviation of the error. The same trend is observed in previous neural network models. Since the output data is much narrower than the original on/off data; therefore, the classification of a positive detection would be expected to have a smaller variation. However, this is a trade-off between the mean absolute timing error and F1-score. We could generalise the observation into that “when the output is more pinpointed, the trained models are more precise in its prediction, but more uncertain to when the actual truth occurs.”

The 5-samples wide transition output signal is a middle ground between the on/off state output and the transition output of its rising and falling edges. It reduces the chance that the model had failed in its training. It also improves the accuracy of a timely detection on the transition; however, the widening from one data sample to five data samples has made the prediction more varying thus increase the standard deviation of the timing error.

Using transition outputs do not produce more timely prediction, although the output is being pinpointed. It is largely due to a large amount of non-transitional output in the training data. Since the non-transitional output heavily outnumbers the transition output, the model recognises itself has a good match with most of the actual ground truth even when it is unresponsive to any input and hold the default output of zero.

The number of unresponsive trained models is reduced when we changed the cost of misclassifying the transitional output five times higher than the non-transitional output. However, this has an undesirable outcome of increasing the false positive rate. The increase in the transition output cost, both single sample and five-sample may not improve the F1-score. It increases both the true positive rate and the false positive rate. It is the consequence of allowing more error to be made from the steady-state output into either of the transition output.

The fundamental challenge here is that most technique developed for machine learning is aimed to reproduce the output signal from learning the historical data. However, in a time-critical application such as gait phase detection, the ideal output is one that could pinpoint the occurrence of the transition between states, not one that matches the number of the output sample the most. Our application does not aim to reproduce the entire output signal; it only interested in predicting timely correct transitions.

This fundamental challenge is evident by examining the accuracy of the training and the testing results, see tables in Appendix C. Accuracy shows the percentage of the predicted signal matches the actual signal in the training dataset. Both activities share a high level of accuracy for the trained models, given most of the predicted output is true positives for steady-state, namely, the number of samples with an output equal to zero. A cost function that minimised all misclassified outputs would focus more on getting these steady-state correct than the targeted transition.

Simply increasing the misclassification cost for the transition output will increase false positive detections using the same trained model, which is also not ideal, as evident in our results. The general machine learning scheme does not align with the objective of our application. A more advanced training scheme should be investigated and developed in the future for time-critical application.

Although none of the trained models outperforms the rule-based algorithm in all aspect: F1-score, timeliness, consistency and usefulness, some models offer a viable alternative to the rule-based approach. The system could deploy any of these models to complement the performance of the rule-based approach. It is possible to have an electronics device with enough memory to hold multiple trained models and target the phases with the best ones. The system would then take the best prediction among the models to enhance its performance.

The selection of the model would depend on the hardware and functional requirement. If hardware requires a fast response, the prediction rate would be a limiting factor. Then, some classifier would not be unsuitable, see the training time in Appendix C. For applications with a higher tolerance in timing error, a model with better F1-score could be selected. For applications that require 95% of all detection to be within 50 ms, models with more than 5% likelihood of sitting outside an absolute timing error of 50 ms would be unsuitable. Readers could use the results from this study as a reference to select the most suitable classifiers among the ones tested here for their application. The training time and prediction rate is subjected to hardware performance. The results would only serve as a comparison guide between the classifiers under the same hardware condition.

Scatter plots of the dataset with transition output supports one of the findings in Chapter 4. It is useful to interpret the IMU measurement into variables such as the body kinematics and the presumed joint angle (the difference in the attached body's tilt angle) for gait phase detection because these variables could describe the phases more accurately. These variables would also benefit the training of classifier to produce a better output.

It is apparent that the clusters for initial contact and foot off are separate from Fig 5.4 and Fig 5.5. For stair ascent, the transition outputs are in two distinct clusters on the axes of thigh kinematics (the tilt angle in the sagittal plane). Similarly, the output clusters are distinct in the axes of shank kinematics (the tilt angle in the sagittal plane) for stair descent. The kinematics of the presumed knee joint is the most robust variable. The transition output clusters are separated on the axes of knee kinematics for both stair ascent and descent. This observation reconfirms that using biomechanical description is a more reliable method to find the pattern in the data for gait phase detection. It is recommended that future research should include biomechanics variables in defining rules and machine learning training.

Figure 5.4 and Figure 5.5 show the scatter plots of the 5-sample wide transition output data. It demonstrates there is a clear separation between the clusters of the two transition outputs, IC and FO. Inclusion of variable that could define a clear separation would hugely benefit the trained model by reducing the misclassifying of one cluster to another. Single sample transition output is a subset of the 5-sample wide output. Therefore, the scatter plots of single sample output are omitted. Meanwhile, there would be an overlapping region between the clusters using on/off state output on any pair of input axes.

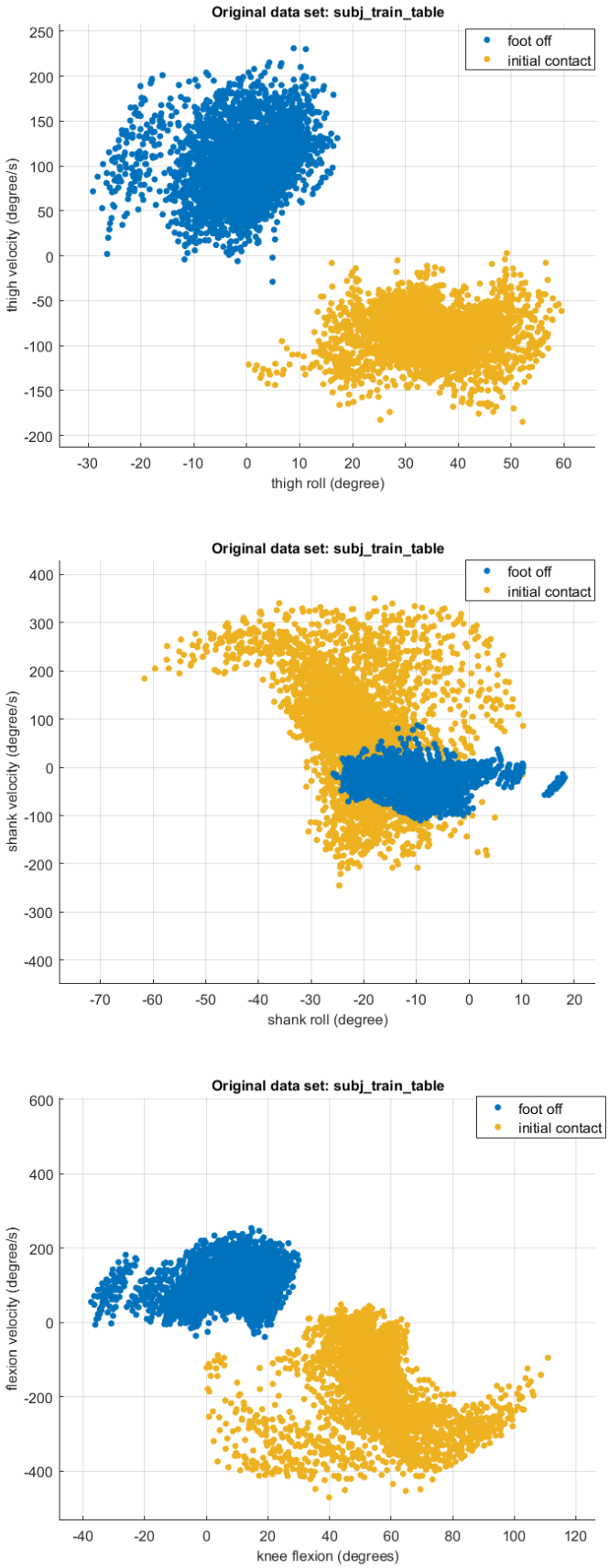


Figure 5.4: Scatter plot of the original data for the two 5-sample wide transition outputs (initial contact and foot off) during stair ascent with axes presenting the angle and velocity of each attached body (top: thigh, middle: shank), and the presumed joint (bottom)

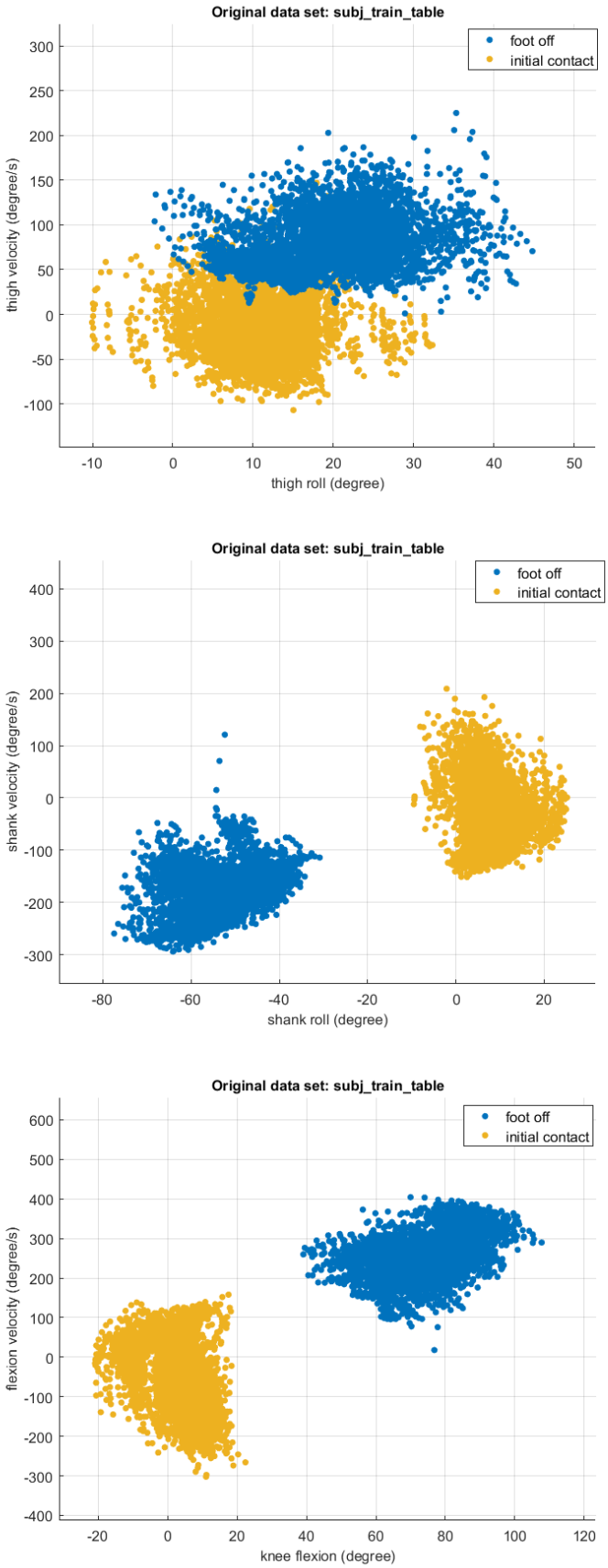


Figure 5.5: Scatter plot of the original data for the two 5-sample wide transition outputs (initial contact and foot off) during stair descent with axes presenting the angle and velocity of each attached body (top: thigh, middle: shank), and the presumed joint (bottom)

5.5 Conclusion

Both machine learning techniques explored in this study show a greater generalisation than the rule-based method presented in Chapter 4. The analysis shows that several trained models outperform the rule-based approach either in F1-score or timing performance.

Most of the improvements over the rule-based approach are made for stair ascent gaits. None of the machine learning approaches yields a lesser likelihood to detect the gait phases transition within 50 ms for stair descent. Hence, the rule-based approach is still better suited for biomechanical applications. It is possible to use the trained models to complement the rule-based method's performance in stair ascent by running them concurrently.

Implementation of the trained models needs to consider the hardware limitation since many of them required either a large onboard memory or high computational power or both. It is not recommended to use budget microcontroller similar to those used in this study, as described in Chapter 3. Researchers and developers are advised to choose a model that is suitable for their application.

Several research gaps have been identified for the tested machine learning technique for the application of GPD. Future research directions could be investigated to address the limitations of the current technique as evidence from the results of this study. Future work will be listed in Chapter 6.

This chapter shows that it is possible to have a trade-off between timing performance and F1-score by transforming the on/off state output to a transition output. The results compare the performance of trained models available in Matlab 2020a/b and present them in a tabular format as a guide. It is up to the developers to choose which approach is better for their application.

CHAPTER 6 : CONCLUSION AND FUTURE WORKS

The last chapter of this study summarises all the findings throughout the previous chapters and the newly discovered gaps resulting from those findings. It also lay a pathway for future research opportunities that would address the gaps identified.

Firstly, we developed an integrated system with both the IMU and footswitches acquisition. The system can operate at the highest output data rate of the IMU at 100 Hz. The sensors' implementation provides the IMUs' tilt angle measurement with an offset no larger than 6 degrees compared to an encoder reference system. One major future research opportunity is to incorporate an auto-labelling program, so the data acquisition is ready for the assessor to evaluate or for machine learning training. Another opportunity is to verify the accuracy of the IMU motion tracking system against the golden standard of optical 3D motion capturing system such as VICON. A verified system would open many opportunities for researchers to study gait over various terrain and realistic environment.

A knowledge gap is found when evaluating sensor performance. Current technique used in normality test, a statistic test to verify if the data is normally distributing, does not recognise a normal distribution that is highly discretised with the tails truncated for the large data sample. These kinds of digital signals are found in systems where the sensor is accurate, and the fluctuation is within a few increments of the resolution. Without a proper technique to verify whether the data fluctuation is normal or not, it is inconclusive to tell if the IMU sensor's bias is compensated or just attenuated after the calibration. A new statistic normality test should be developed for evaluating consistent digital data.

The developed rule-based detection has been implemented and verified on a physical device. The performance achieves a high overall F1-score of 0.9925 with a mean error [standard deviation] of 43.25[30.21], 20.12[15.23], -30.17[23.43], and -43.66[16.41] ms for ascent IC, descent IC, ascent EC, and descent EC respectively. Their 95% CI are: [-16.17,102.67], [-76.62,15.22], [-9.73,49.97], and [-75.82, -11.50] respectively before offsetting the early detection. Ascent IC and ascent EC could be offset to have a 95% CI of [-59.43,59.43], and [-29.85, 29.85] respectively. The mean is below 50ms on average which is the target tolerance for biomechanical application, over 3419 steady-state steps across 21 healthy participants for both stair ascent (1665 steps: 1916 initial contact and 1665 end contact) and descent ascent (1754 steps: 2027 initial contact and 1754 end contact) gait. It

shows that incorporating the biomechanics definition of the gait phases in defining rules for detection is a practical and reliable method in developing a detection algorithm. This study conducted the experiment in the exact operational environment of a knee assistive device, which assists a user in performing multiple progressive steps on a staircase. The performance presented should be reflective to the practical performance in detecting the stance and swing phase of the user when it is deployed. Therefore, the study proves the readiness of implementing the technology on similar knee devices.

This study shows a rule-based approach with an accurate definition based on relevant knowledge could reliably detecting gait phases. The results support existing biomechanics reports that stair gait has a much higher inter-subject and intra-subject variation among people within the same demography compared to level walking. It is indicated by the large standard deviation of the timing error, particularly in stair ascent. The standard deviation of the reported detection would mean that 18.63% of all detection would sit outside the 50 ms tolerance. This intrinsic variation remains the biggest hurdle for developing a robust detection for stair ambulation. From our data, the initial contact of stair ascent could occur between two kinematic events which are the minimum of knee flexion velocity and the minimum of shank tilt velocity. This implies that there are different strategies for stair ambulation. Several future research opportunities had been identified. A higher level of classification could detect and facilitate a tailored rule-based algorithm for each gait variation. This may lead to biomechanics studies that observe, classify and explain the different stair ambulation strategies. The study could also extend to other gait activities.

The study found that using body kinematics such as the YPR angles of the attached segments, and the difference in their tilt angles (the presumed joint angle between the attached segment) are useful parameters for deriving both detection rules and for machine learning training. It allows translating the biomechanics observation into mathematical rules. This study would encourage using these biomechanically significant variables to be the basis of defining detection rules and training machine learning models. Interpreting the data into the limb's physical orientation would allow biomechanists to study the gait outside the laboratory environment. This could also strengthen the collaboration between the field of biomechanics and engineering, where the results could be examined by both areas.

Both machine learning techniques explored in this study could improve the rule-based approach with concurrent deployment. Multiple trained models are identified to have a better

performance in either F1-score or timing error. The generalisation of the machine learning approaches is better than the rule-based one. However, none of the trained models can outperform the rule-based approach's timing performance during stair descent gait. Modification to the training scheme and the model's architecture may be required to better suit the application of detecting gait phases.

The study offers a comparison between each of the common machine learning techniques such as supervised learning classifiers, NIO and NARX network available in MATLAB 2020 a/b. The results could serve as a guide that compares the training and testing performance of the above techniques. It is advised that developers of GPD select the most suitable ones within their hardware limitations.

Time-series neural network could experiment with multiple outputs to enhance the ability to generate a discrete output signal with less ambiguity between each output values. The input data could be transformed into a format so that technique from other active fields of machine learning could be deployed, such as image process techniques and transformer network. Transfer learning could be a focus in future research to train a network for a different activity quickly.

A future investigation could produce a more accurate representation of the output signal rather than transform the predicted signal with a filter. One method would be minimising the integration of the absolute difference between the expected output and the training output. The first derivative of the difference could be incorporated as part of the cost function during training to make the transition between the output level more apparent, and reduce the effort spent finding the optimal filtering for the predicted signal.

Another method that may be suitable is a Transformer network [171] and other natural language processing (NLP) algorithms. Time-series data could be treated as an array of data, similar to NLP, to interpret the information from this sequential input. Similar to how existing programs could interpret the emotional tone from text using machine learning [172], we could train the IMU measurement to interpret the phases. Another advantage that the transformer network could offer is the possible transferability in its learning. Transformer networks use an "attention mechanism" [173], which encode both the input and output and then tune the correlation between them in the encrypted format. The resultant trained network may be suitable for transfer learning between different gaits or gait strategies, for

example, healthy and pathological gaits. This could solve the issue of transferability for a trained network in detecting phases on different target demographics.

Furthermore, an alternative output preparation method that would retain the option to trade-off F1-score with timing performance using transition output while potentially allowing more than three output values. The network structure would be changed to allow multiple output signals of ones and zeros, each output representing the positive and negatives of a specific event. This architecture is drawn from the method being used for object identification on images. In a typical image-based object identification network, the network is trained to identify and locate the targeted object on the image. Sometimes the network can be trained to identify multiple objects. In that case, the network is trained to have an array of output with each element on the array representing the score of identifying each specific object. A score of 1 indicates that the network is certain of the identification. The final labelling of each region of the picture will be the element with the highest value and above a certain threshold.

In the application of gait phase detection, a similar network would have multiple time-series outputs. Each output would identify the occurrence of a specific event. Multiple outputs would allow the prediction of each event to be made independent without the possible ambiguity from other events while maintaining the improved timing trade-off of using a transition only output type. The trade-off of multiple output training would require a much larger working memory and a much longer training time. The construction and testing of this new network for GPD should be one of the future studies.

Supervised learning classifiers could be enhanced by modifying the training scheme and the cost function to consider the timing performance. Unsupervised or reinforcement learning could be deployed to find hidden clusters and the ability to learn from new data input.

One method that could improve the training is to give a score for the time difference between the rising and falling edge of the labelled data and those of the prediction; therefore, the resultant classifier may be optimised for time-critical application such as gait phase detection. The cost function should be modified to include the timing score. The optimisation could then search for a trained model that can minimise the timing difference in the transition between the prediction and the ground truth. This training option is not readily available in current machine learning toolbox in many software platforms. Future studies should focus

on developing a new architecture to better train machine learning classifiers for timing performance in their classification.

Another method that could improve the prediction of the transition is combining the training with unsupervised or reinforcement learning techniques. The model would have the ability to learn new clusters and minimise the cost function by evaluating its' previous output to the ground truth.

At last, the study had fulfilled its primary goal and deliver a GPD algorithm for detecting gait phases during stair ambulation. The outcome is implemented on a physical wearable knee brace and is proven in its intended operational environment. The observation made from the analysis also opens the door to further scientific investigation and biomechanics and machine learning outcomes.

REFERENCE

[1] J. E. Pratt, B. T. Krupp, C. J. Morse, and S. H. Collins, "The RoboKnee: an exoskeleton for enhancing strength and endurance during walking," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*, 2004, vol. 3, pp. 2430-2435: IEEE

[2] A. Chu, H. Kazerooni, and A. Zoss, "On the biomimetic design of the berkeley lower extremity exoskeleton (BLEEX)," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, 2005, pp. 4345-4352: IEEE

[3] F. Gazzani, A. Fadda, M. Torre, and V. Macellari, "WARD: a pneumatic system for body weight relief in gait rehabilitation," *IEEE Transactions on Rehabilitation Engineering,* vol. 8, no. 4, pp. 506-513, 2000.

[4] D. Aoyagi, W. E. Ichinose, S. J. Harkema, D. J. Reinkensmeyer, and J. E. Bobrow, "A robot and control algorithm that can synchronously assist in naturalistic motion during body-weight-supported gait training following neurologic injury," *IEEE Transactions on neural systems and rehabilitation engineering,* vol. 15, no. 3, pp. 387-400, 2007.

[5] G. Zeilig, H. Weingarden, M. Zwecker, I. Dudkiewicz, A. Bloch, and A. Esquenazi, "Safety and tolerance of the ReWalk™ exoskeleton suit for ambulation by people with complete spinal cord injury: A pilot study," *The journal of spinal cord medicine,* vol. 35, no. 2, pp. 96-101, 2012.

[6] A. De Santis, B. Siciliano, A. De Luca, and A. Bicchi, "An atlas of physical human–robot interaction," *Mechanism and Machine Theory,* vol. 43, no. 3, pp. 253-270, 2008.

[7] D. Sanz-Merodio, M. Cestari, J. C. Arevalo, X. Carrillo, and E. Garcia, "Development of a Lower-Limb Active Orthosis and a Walker for Gait Assistance," in *ROBOT2013: First Iberian Robotics Conference*, 2014, pp. 219-233: Springer

[8] P. D. Neuhaus, J. H. Noorden, T. J. Craig, T. Torres, J. Kirschbaum, and J. E. Pratt, "Design and evaluation of Mina: A robotic orthosis for paraplegics," in *2011 IEEE international conference on rehabilitation robotics*, 2011, pp. 1-8: IEEE

[9] S. Wang, C. Meijneke, and H. van der Kooij, "Modeling, design, and optimization of Mindwalker series elastic joint," in *2013 IEEE 13th International Conference on Rehabilitation Robotics (ICORR)*, 2013, pp. 1-8: IEEE

[10] A. Esquenazi and A. Packel, "Robotic-assisted gait training and restoration," *American journal of physical medicine & rehabilitation,* vol. 91, no. 11, pp. S217-S231, 2012.

[11] G. Barbareschi, R. Richards, M. Thornton, T. Carlson, and C. Holloway, "Statically vs dynamically balanced gait: Analysis of a robotic exoskeleton compared with a human," in *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2015, pp. 6728-6731: IEEE

[12] N. Tabti, M. Kardofaki, S. Alfayad, Y. Chitour, F. B. Ouezdou, and E. Dychus, "A Brief Review of the Electronics, Control System Architecture, and Human Interface for Commercial Lower Limb Medical Exoskeletons Stabilized by Aid of Crutches," in *2019 28th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, 2019, pp. 1-6: IEEE.https://doi.org/10.1109/ro-man46459.2019.8956311

[13] C. Liang and T. Hsiao, "Walking strategies and performance evaluation for human-exoskeleton systems under admittance control," *Sensors,* vol. 20, no. 15, p. 4346, 2020.

[14] T. Zhang, M. Tran, and H. Huang, "Admittance shaping-based assistive control of SEA-driven robotic hip exoskeleton," *IEEE/ASME Transactions on Mechatronics,* vol. 24, no. 4, pp. 1508-1519, 2019.

[15] T. Yan, M. Cempini, C. M. Oddo, and N. Vitiello, "Review of assistive strategies in powered lower-limb orthoses and exoskeletons," *Rob. Auton. Syst.,* vol. 64, pp. 120-136, 2015.

[16] A. J. Young and D. P. Ferris, "State of the art and future directions for lower limb robotic exoskeletons," *IEEE Transactions on Neural Systems and Rehabilitation Engineering,* vol. 25, no. 2, pp. 171-182, 2016.

[17] Y. Sankai, "Leading edge of cybernics: Robot suit hal," in *2006 SICE-ICASE International Joint Conference*, 2006, pp. P-1-P-2: IEEE

[18] A. B. Zoss, H. Kazerooni, and A. Chu, "Biomechanical design of the Berkeley lower extremity exoskeleton (BLEEX)," *IEEE/ASME Trans. Mechatronics,* vol. 11, no. 2, pp. 128-138, 2006.
[19] Y. Sankai, "HAL: Hybrid assistive limb based on cybernics," in *Robotics research*: Springer, 2010, pp. 25-34.
[20] M. a. Markets, "Medical Robots Market By Product & Service [Instrument & Accessories, Robotic Systems (Surgical Robots, Rehabilitation Robots)], Application (Laparoscopy, Radiation Therapy, Pharmacy),End User (Hospital, Ambulatory Surgery)-Global Forecast To 2025," Market Research Report, 2020, p. 191. [Online]. Available: https://www.marketsandmarkets.com/Market-Reports/medical-robotic-systems-market-2916860.html.
[21] B. Chen, B. Zi, Z. Wang, L. Qin, and W.-H. Liao, "Knee exoskeletons for gait rehabilitation and human performance augmentation: A state-of-the-art," *Mechanism and Machine Theory,* vol. 134, pp. 499-511, 2019.
[22] P. Beyl *et al.*, "Safe and compliant guidance by a powered knee exoskeleton for robot-assisted rehabilitation of gait," *Advanced Robotics,* vol. 25, no. 5, pp. 513-535, 2011.
[23] E. Papi, D. Osei-Kuffour, Y.-M. A. Chen, and A. H. McGregor, "Use of wearable technology for performance assessment: a validation study," *Medical engineering & physics,* vol. 37, no. 7, pp. 698-704, 2015.
[24] F. Tian, M. S. Hefzy, and M. Elahinia, "State of the art review of Knee–Ankle–Foot orthoses," *Ann. Biomed. Eng.,* vol. 43, no. 2, pp. 427-441, 2015.
[25] A. Esquenazi, M. Talaty, A. Packel, and M. Saulino, "The ReWalk powered exoskeleton to restore ambulatory function to individuals with thoracic-level motor-complete spinal cord injury," *Am. J. Phys. Med. Rehabil.,* vol. 91, no. 11, pp. 911-921, 2012.
[26] K. Suzuki, G. Mito, H. Kawamoto, Y. Hasegawa, and Y. Sankai, "Intention-based walking support for paraplegia patients with Robot Suit HAL," *Adv. Robotics,* vol. 21, no. 12, pp. 1441-1469, 2007.
[27] A. Litwic, M. Edwards, E. Dennison, and C. Cooper, "Epidemiology and Burden of Osteoarthritis. 2013," ed
[28] Arthritis and O. Victoria, *A Problem Worth Solving: The Rising Cost of Musculoskeletal Conditions in Australia: a Report*. Arthritis and Osteoporosis Victoria, 2013
[29] G. Li, T. Liu, and J. Yi, "Wearable sensor system for detecting gait parameters of abnormal gaits: A feasibility study," *IEEE Sensors Journal,* vol. 18, no. 10, pp. 4234-4241, 2018.
[30] J. F. Veneman, R. Kruidhof, E. E. Hekman, R. Ekkelenkamp, E. H. Van Asseldonk, and H. Van Der Kooij, "Design and evaluation of the LOPES exoskeleton robot for interactive gait rehabilitation," *IEEE Trans. Neural Syst. Rehabil. Eng,* vol. 15, no. 3, pp. 379-386, 2007.
[31] S. K. Banala, S. H. Kim, S. K. Agrawal, and J. P. Scholz, "Robot assisted gait training with active leg exoskeleton (ALEX)," *IEEE transactions on neural systems and rehabilitation engineering,* vol. 17, no. 1, pp. 2-8, 2008.
[32] C. Hartigan *et al.*, "Mobility outcomes following five training sessions with a powered exoskeleton," *Topics in spinal cord injury rehabilitation,* vol. 21, no. 2, pp. 93-99, 2015.
[33] E. Squyer, D. L. Stamper, D. T. Hamilton, J. A. Sabin, and S. S. Leopold, "Unloader knee braces for osteoarthritis: do patients actually wear them?," *Clin. Orthop. Relat. Res.,* vol. 471, no. 6, pp. 1982-1991, 2013.
[34] J. Taborri, E. Palermo, S. Rossi, and P. Cappa, "Gait partitioning methods: A systematic review," *Sensors,* vol. 16, no. 1, p. 66, 2016.
[35] R. Caldas, M. Mundt, W. Potthast, F. B. de Lima Neto, and B. Markert, "A systematic review of gait analysis methods based on inertial sensors and adaptive algorithms," *Gait posture,* vol. 57, pp. 204-210, 2017.
[36] C. M. O'Connor, S. K. Thorpe, M. J. O'Malley, and C. L. Vaughan, "Automatic detection of gait events using kinematic data," *Gait posture,* vol. 25, no. 3, pp. 469-474, 2007.
[37] M. S. Aung *et al.*, "Automated detection of instantaneous gait events using time frequency analysis and manifold embedding," *IEEE Trans. Neural Syst. Rehabil. Eng,* vol. 21, no. 6, pp. 908-916, 2013.
[38] D. T. Felson, "Challenges of identifying and treating patellofemoral osteoarthritis," ed: BMJ Publishing Group Ltd and British Association of Sport and Exercise Medicine, 2016

[39] M. G. Wheatley, M. J. Rainbow, and A. L. Clouthier, "Patellofemoral mechanics: a review of pathomechanics and research approaches," *Current reviews in musculoskeletal medicine,* vol. 13, pp. 326-337, 2020.
[40] Y.-M. Kim and Y.-B. Joo, "Patellofemoral osteoarthritis," *Knee surgery & related research,* vol. 24, no. 4, p. 193, 2012.
[41] K. M. Crossley *et al.*, "2016 Patellofemoral pain consensus statement from the 4th International Patellofemoral Pain Research Retreat, Manchester. Part 1: Terminology, definitions, clinical examination, natural history, patellofemoral osteoarthritis and patient-reported outcome measures," *British Journal of Sports Medicine,* vol. 50, no. 14, pp. 839-843, 2016.
[42] K. M. Crossley, M. van Middelkoop, M. J. Callaghan, N. J. Collins, M. S. Rathleff, and C. J. Barton, "2016 Patellofemoral pain consensus statement from the 4th International Patellofemoral Pain Research Retreat, Manchester. Part 2: recommended physical interventions (exercise, taping, bracing, foot orthoses and combined interventions)," *British journal of sports medicine,* vol. 50, no. 14, pp. 844-852, 2016.
[43] H.-P. W. Van Jonbergen, R. W. Poolman, and A. van Kampen, "Isolated patellofemoral osteoarthritis: a systematic review of treatment options using the GRADE approach," *Acta orthopaedica,* vol. 81, no. 2, pp. 199-205, 2010.
[44] R. J. Foster, A. R. De Asha, N. D. Reeves, C. N. Maganaris, and J. G. Buckley, "Stair-specific algorithms for identification of touch-down and foot-off when descending or ascending a non-instrumented staircase," *Gait posture,* vol. 39, no. 2, pp. 816-821, 2014.
[45] R. C. González, A. M. López, J. Rodriguez-Uría, D. Alvarez, and J. C. Alvarez, "Real-time gait event detection for normal subjects from lower trunk accelerations," *Gait posture,* vol. 31, no. 3, pp. 322-325, 2010.
[46] M. Roerdink, C. J. Lamoth, and P. J. Beek, "Online gait event detection using a large force platform embedded in a treadmill," *J. Biomech.,* vol. 41, no. 12, pp. 2628-2632, 2008.
[47] P. Merriaux, Y. Dupuis, R. Boutteau, P. Vasseur, and X. Savatier, "A study of vicon system positioning performance," *Sensors,* vol. 17, no. 7, p. 1591, 2017.
[48] J. L. McGinley, R. Baker, R. Wolfe, and M. E. Morris, "The reliability of three-dimensional kinematic gait measurements: a systematic review," *Gait posture,* vol. 29, no. 3, pp. 360-369, 2009.
[49] C. D. Joshi, U. Lahiri, and N. V. Thakor, "Classification of gait phases from lower limb EMG: Application to exoskeleton orthosis," in *Point-of-Care Healthcare Technologies (PHT), 2013 IEEE*, 2013, pp. 228-231: IEEE.https://doi.org/10.1109/pht.2013.6461326
[50] R. T. Lauer, B. T. Smith, and R. R. Betz, "Application of a neuro-fuzzy network for gait event detection using electromyography in the child with cerebral palsy," *IEEE Trans. Biomed. Eng.,* vol. 52, no. 9, pp. 1532-1540, 2005.
[51] R. T. Lauer, B. T. Smith, D. Coiro, R. R. Betz, and J. McCarthy, "Feasibility of gait event detection using intramuscular electromyography in the child with cerebral palsy," *Neuromodulation: Technology at the Neural Interface,* vol. 7, no. 3, pp. 205-213, 2004.
[52] P. Catalfamo, D. Moser, S. Ghoussayni, and D. Ewins, "Detection of gait events using an F-Scan in-shoe pressure measurement system," *Gait posture,* vol. 28, no. 3, pp. 420-426, 2008.
[53] B. T. Smith, D. J. Coiro, R. Finson, R. R. Betz, and J. McCarthy, "Evaluation of force-sensing resistors for gait event detection to trigger electrical stimulation to improve walking in the child with cerebral palsy," *IEEE Trans. Neural Syst. Rehabil. Eng,* vol. 10, no. 1, pp. 22-29, 2002.
[54] M. Hanlon and R. Anderson, "Real-time gait event detection using wearable sensors," *Gait posture,* vol. 30, no. 4, pp. 523-527, 2009.
[55] H. F. Maqbool, M. A. B. Husman, M. I. Awad, A. Abouhossein, N. Iqbal, and A. A. Dehghani-Sanij, "A real-time gait event detection for lower limb prosthesis control and evaluation," *IEEE Trans. Neural Syst. Rehabil. Eng,* vol. 25, no. 9, pp. 1500-1509, 2017.
[56] A. T. M. Willemsen, F. Bloemhof, and H. B. Boom, "Automatic stance-swing phase detection from accelerometer data for peroneal nerve stimulation," *IEEE Trans. Biomed. Eng.,* vol. 37, no. 12, pp. 1201-1208, 1990.

[57] A. M. Sabatini, C. Martelloni, S. Scapellato, and F. Cavallo, "Assessment of walking features from foot inertial sensing," *IEEE Trans. Biomed. Eng.,* vol. 52, no. 3, pp. 486-494, 2005.
[58] M. M. Skelly and H. J. Chizeck, "Real-time gait event detection for paraplegic FES walking," *IEEE Trans. Neural Syst. Rehabil. Eng,* vol. 9, no. 1, pp. 59-68, 2001.
[59] D. Kotiadis, H. J. Hermens, and P. H. Veltink, "Inertial Gait Phase Detection for control of a drop foot stimulator: Inertial sensing for gait phase detection," *Med. Eng. Phys.,* vol. 32, no. 4, pp. 287-297, 2010.
[60] P. Catalfamo, S. Ghoussayni, and D. Ewins, "Gait event detection on level ground and incline walking using a rate gyroscope," *Sensors,* vol. 10, no. 6, pp. 5683-5702, 2010.
[61] J. Rueterbories, E. G. Spaich, B. Larsen, and O. K. Andersen, "Methods for gait event detection and analysis in ambulatory systems," *Med. Eng. Phys.,* vol. 32, no. 6, pp. 545-552, 2010.
[62] A. Behboodi, N. Zahradka, H. Wright, J. Alesi, and S. Lee, "Real-time detection of seven phases of gait in children with cerebral palsy using two gyroscopes," *Sensors,* vol. 19, no. 11, p. 2517, 2019.
[63] H.-C. Chang, Y.-L. Hsu, S.-C. Yang, J.-C. Lin, and Z.-H. Wu, "A wearable inertial measurement system with complementary filter for gait analysis of patients with stroke or Parkinson's disease," *IEEE Access,* vol. 4, pp. 8442-8453, 2016.
[64] X. Meng, H. Yu, and M. P. Tham, "Gait phase detection in able-bodied subjects and dementia patients," in *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2013, pp. 4907-4910: IEEE
[65] N. C. Bejarano, E. Ambrosini, A. Pedrocchi, G. Ferrigno, M. Monticone, and S. Ferrante, "A novel adaptive, real-time algorithm to detect gait events from wearable sensors," *IEEE transactions on neural systems and rehabilitation engineering,* vol. 23, no. 3, pp. 413-422, 2014.
[66] J. K. Lee and E. J. Park, "Quasi real-time gait event detection using shank-attached gyroscopes," *Med. Biol. Eng. Comput.,* vol. 49, no. 6, pp. 707-712, 2011.
[67] A. Sant'Anna and N. Wickström, "A symbol-based approach to gait analysis from acceleration signals: Identification and detection of gait events and a new measure of gait symmetry," *IEEE Trans. Inf. Technol. Biomed.,* vol. 14, no. 5, pp. 1180-1187, 2010.
[68] K. Aminian, B. Najafi, C. Büla, P.-F. Leyvraz, and P. Robert, "Spatio-temporal parameters of gait measured by an ambulatory system using miniature gyroscopes," *J. Biomech.,* vol. 35, no. 5, pp. 689-699, 2002.
[69] H. Lakany, "Extracting a diagnostic gait signature," *Pattern recognition,* vol. 41, no. 5, pp. 1627-1637, 2008.
[70] H. Zhou *et al.*, "Towards real-time detection of gait events on different terrains using time-frequency analysis and peak heuristics algorithm," *Sensors,* vol. 16, no. 10, p. 1634, 2016.
[71] E. D. Ledoux, "Inertial sensing for gait event detection and transfemoral prosthesis control strategy," *IEEE Transactions on Biomedical Engineering,* vol. 65, no. 12, pp. 2704-2712, 2018.
[72] S. Khandelwal and N. Wickström, "Gait event detection in real-world environment for long-term applications: Incorporating domain knowledge into time-frequency analysis," *IEEE transactions on neural systems and rehabilitation engineering,* vol. 24, no. 12, pp. 1363-1372, 2016.
[73] A. Parri *et al.*, "Real-time hybrid locomotion mode recognition for lower limb wearable robots," *IEEE/ASME Trans. Mechatronics,* vol. 22, no. 6, pp. 2480-2491, 2017.
[74] J. Lu and E. Zhang, "Gait recognition for human identification based on ICA and fuzzy SVM through multiple views fusion," *Pattern Recognition Letters,* vol. 28, no. 16, pp. 2401-2411, 2007.
[75] J.-H. Yoo, D. Hwang, and M. S. Nixon, "Gender classification in human gait using support vector machine," in *International Conference on Advanced Concepts for Intelligent Vision Systems*, 2005, pp. 138-145: Springer
[76] H.-y. Lau, K.-y. Tong, and H. Zhu, "Support vector machine for classification of walking conditions of persons after stroke with dropped foot," *Hum. Movement Sci.,* vol. 28, no. 4, pp. 504-514, 2009.

[77] E. Zheng, S. Manca, T. Yan, A. Parri, N. Vitiello, and Q. Wang, "Gait phase estimation based on noncontact capacitive sensing and adaptive oscillators," *IEEE Trans. Biomed. Eng.,* vol. 64, no. 10, pp. 2419-2430, 2017.
[78] B. Hu, E. Rouse, and L. Hargrove, "Fusion of bilateral lower-limb neuromechanical signals improves prediction of locomotor activities," *Frontiers in Robotics and AI,* vol. 5, p. 78, 2018.
[79] G. Chen, P. Qi, Z. Guo, and H. Yu, "Gait-event-based synchronization method for gait rehabilitation robots via a bioinspired adaptive oscillator," *IEEE Trans. Biomed. Eng.,* vol. 64, no. 6, pp. 1345-1356, 2016.
[80] J. Bae and M. Tomizuka, "Gait phase analysis based on a Hidden Markov Model," *Mechatronics,* vol. 21, no. 6, pp. 961-970, 2011.
[81] S. Crea *et al.*, "Development of gait segmentation methods for wearable foot pressure sensors," in *2012 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2012, pp. 5018-5021: IEEE
[82] A. Mannini and A. M. Sabatini, "Gait phase detection and discrimination between walking–jogging activities using hidden Markov models applied to foot motion data from a gyroscope," *Gait & posture,* vol. 36, no. 4, pp. 657-661, 2012.
[83] A. Mannini, V. Genovese, and A. M. Sabatini, "Online decoding of hidden Markov models for gait event detection using foot-mounted gyroscopes," *IEEE journal of biomedical and health informatics,* vol. 18, no. 4, pp. 1122-1130, 2013.
[84] M. D. Sánchez Manchola, M. J. P. Bernal, M. Munera, and C. A. Cifuentes, "Gait phase detection for lower-limb exoskeletons using foot motion data from a single inertial measurement unit in hemiparetic individuals," *Sensors,* vol. 19, no. 13, p. 2988, 2019.
[85] K. Kong and M. Tomizuka, "A gait monitoring system based on air pressure sensors embedded in a shoe," *IEEE/ASME Transactions on mechatronics,* vol. 14, no. 3, pp. 358-370, 2009.
[86] M. Alaqtash, H. Yu, R. Brower, A. Abdelgawad, and T. Sarkodie-Gyan, "Application of wearable sensors for human gait analysis using fuzzy computational algorithm," *Engineering Applications of Artificial Intelligence,* vol. 24, no. 6, pp. 1018-1025, 2011.
[87] A. Alvarez-Alvarez, G. Trivino, and O. Cordon, "Human gait modeling using a genetic fuzzy finite state machine," *IEEE Transactions on Fuzzy Systems,* vol. 20, no. 2, pp. 205-223, 2011.
[88] A. Miller, "Gait event detection using a multilayer neural network," *Gait posture,* vol. 29, no. 4, pp. 542-545, 2009.
[89] R. Williamson and B. J. Andrews, "Gait event detection for FES using accelerometers and supervised machine learning," *IEEE Transactions on Rehabilitation Engineering,* vol. 8, no. 3, pp. 312-319, 2000.
[90] R. L. Evans and D. Arvind, "Detection of gait phases using orient specks for mobile clinical gait analysis," in *2014 11th International Conference on Wearable and Implantable Body Sensor Networks*, 2014, pp. 149-154: IEEE
[91] D.-X. Liu, X. Wu, W. Du, C. Wang, and T. Xu, "Gait phase recognition for lower-limb exoskeleton with only joint angular sensors," *Sensors,* vol. 16, no. 10, p. 1579, 2016.
[92] J.-Y. Jung, W. Heo, H. Yang, and H. Park, "A neural network-based gait phase classification method using sensors equipped on lower limb exoskeleton robots," *Sensors,* vol. 15, no. 11, pp. 27738-27759, 2015.
[93] T. Zhen, L. Yan, and P. Yuan, "Walking gait phase detection based on acceleration signals using LSTM-DNN Algorithm," *Algorithms,* vol. 12, no. 12, p. 253, 2019.
[94] M. Islam and E. T. Hsiao-Wecksler, "Detection of gait modes using an artificial neural network during walking with a powered ankle-foot orthosis," *Journal of Biophysics,* vol. 2016, 2016.
[95] I. P. Pappas, M. R. Popovic, T. Keller, V. Dietz, and M. Morari, "A reliable gait phase detection system," *IEEE Trans. Neural Syst. Rehabil. Eng,* vol. 9, no. 2, pp. 113-125, 2001.
[96] P. C. Formento, R. Acevedo, S. Ghoussayni, and D. Ewins, "Gait event detection during stair walking using a rate gyroscope," *Sensors,* vol. 14, no. 3, pp. 5470-5485, 2014.

[97] B. Coley, B. Najafi, A. Paraschiv-Ionescu, and K. Aminian, "Stair climbing detection during daily physical activity using a miniature gyroscope," *Gait posture,* vol. 22, no. 4, pp. 287-294, 2005.
[98] L. V. Ojeda *et al.*, "Estimating Stair Running Performance Using Inertial Sensors," *Sensors,* vol. 17, no. 11, p. 2647, 2017.
[99] J. Yang *et al.*, "Machine Learning Based Adaptive Gait Phase Estimation Using Inertial Measurement Sensors," in *2019 Design of Medical Devices Conference*, 2019: American Society of Mechanical Engineers Digital Collection.https://doi.org/10.1115/dmd2019-3266
[100] H. T. T. Vu *et al.*, "A review of gait phase detection algorithms for lower limb prostheses," *Sensors,* vol. 20, no. 14, p. 3972, 2020.
[101] A. Khan and E. Biddiss, "Musical Stairs: A motivational therapy tool for children with disabilities featuring automated detection of stair-climbing gait events via inertial sensors," *Med. Eng. Phys.,* vol. 40, pp. 95-102, 2017.
[102] C. M. Archer, J. Lach, S. Chen, M. F. Abel, and B. C. Bennett, "Activity classification in users of ankle foot orthoses," *Gait posture,* vol. 39, no. 1, pp. 111-117, 2014.
[103] D. Levine, J. Richards, and M. W. Whittle, *Whittle's Gait Analysis-E-Book*. Elsevier health sciences, 2012
[104] B. J. McFadyen and D. A. Winter, "An integrated biomechanical analysis of normal stair ascent and descent," *J. Biomech.,* vol. 21, no. 9, pp. 733-744, 1988.
[105] A. Protopapadaki, W. I. Drechsler, M. C. Cramp, F. J. Coutts, and O. M. Scott, "Hip, knee, ankle kinematics and kinetics during stair ascent and descent in healthy young individuals," *Clinical biomechanics,* vol. 22, no. 2, pp. 203-210, 2007.
[106] S. J. Abbas and Z. M. Abdulhassan, "Kinematic analysis of human climbing up and down stairs at different inclinations," *Eng Tech Journal, Part A,* vol. 31, no. 8, pp. 1556-1566, 2013.
[107] T. Cluff and D. G. E. Robertson, "Kinetic analysis of stair descent: Part 1. Forwards step-over-step descent," *Gait & posture,* vol. 33, no. 3, pp. 423-428, 2011.
[108] Y.-C. Lin, L. A. Fok, A. G. Schache, and M. G. Pandy, "Muscle coordination of support, progression and balance during stair ambulation," *J. Biomech.,* vol. 48, no. 2, pp. 340-347, https://doi.org/10.1016/j.jbiomech.2014.11.019 2015.
[109] J. Lewis, G. Freisinger, X. Pan, R. Siston, L. Schmitt, and A. Chaudhari, "Changes in lower extremity peak angles, moments and muscle activations during stair climbing at different speeds," *Journal of Electromyography and Kinesiology,* vol. 25, no. 6, pp. 982-989, 2015.
[110] R. Riener, M. Rabuffetti, and C. Frigo, "Stair ascent and descent at different inclinations," *Gait posture,* vol. 15, no. 1, pp. 32-44, 2002.
[111] S. M. Reid, S. K. Lynn, R. P. Musselman, and P. A. Costigan, "Knee biomechanics of alternate stair ambulation patterns," *Medicine and science in sports and exercise,* vol. 39, no. 11, pp. 2005-2011, 2007.
[112] L. A. Livingston, J. M. Stevenson, and S. J. Olney, "Stairclimbing kinematics on stairs of differing dimensions," *Archives of physical medicine and rehabilitation,* vol. 72, no. 6, pp. 398-402, 1991.
[113] A. C. Novak, V. Komisar, B. E. Maki, and G. R. Fernie, "Age-related differences in dynamic balance control during stair descent and effect of varying step geometry," *Applied ergonomics,* vol. 52, pp. 275-284, 2016.
[114] H. Iijima, K. Shimoura, T. Aoyama, and M. Takahashi, "Biomechanical characteristics of stair ambulation in patients with knee OA: a systematic review with meta-analysis toward a better definition of clinical hallmarks," *Gait & posture,* vol. 62, pp. 191-201, 2018.
[115] T. Schmalz, S. Blumentritt, and B. Marx, "Biomechanical analysis of stair ambulation in lower limb amputees," *Gait & posture,* vol. 25, no. 2, pp. 267-278, 2007.
[116] I. Bosse, K. D. Oberländer, H. H. Savelberg, K. Meijer, G.-P. Brüggemann, and K. Karamanidis, "Dynamic stability control in younger and older adults during stair descent," *Human movement science,* vol. 31, no. 6, pp. 1560-1570, 2012.
[117] R. Brumbaugh, R. Crowninshield, W. Blair, and J. Andrews, "An in-vivo study of normal wrist kinematics," 1982.
[118] E. Y. Chao, "Justification of triaxial goniometer for the measurement of joint rotation," *Journal of Biomechanics,* vol. 13, no. 12, pp. 989-1006, 1980.

[119] W. C. Hayes, J. Gran, M. L. Nagurka, J. Feldman, and C. Oatis, "Leg motion analysis during gait by multiaxial accelerometry: theoretical foundations and preliminary validations," 1983.
[120] A. Cappozzo, "Gait analysis methodology," *Human movement science,* vol. 3, no. 1-2, pp. 27-50, 1984.
[121] A. Cappozzo, A. Cappello, U. D. Croce, and F. Pensalfini, "Surface-marker cluster design criteria for 3-D bone movement reconstruction," *IEEE Transactions on Biomedical Engineering,* vol. 44, no. 12, pp. 1165-1174, 1997.
[122] R. B. Davis III, S. Ounpuu, D. Tyburski, and J. R. Gage, "A gait analysis data collection and reduction technique," *Human movement science,* vol. 10, no. 5, pp. 575-587, 1991.
[123] M. P. Kadaba, H. Ramakrishnan, and M. Wootten, "Measurement of lower extremity kinematics during level walking," *Journal of orthopaedic research,* vol. 8, no. 3, pp. 383-392, 1990.
[124] B. E. Groen, M. Geurts, B. Nienhuis, and J. Duysens, "Sensitivity of the OLGA and VCM models to erroneous marker placement: Effects on 3D-gait kinematics," *Gait & posture,* vol. 35, no. 3, pp. 517-521, 2012.
[125] G. Wu *et al.*, "ISB recommendation on definitions of joint coordinate system of various joints for the reporting of human joint motion—part I: ankle, hip, and spine," *J. Biomech.,* vol. 35, no. 4, pp. 543-548, 2002.
[126] G. Wu *et al.*, "ISB recommendation on definitions of joint coordinate systems of various joints for the reporting of human joint motion—Part II: shoulder, elbow, wrist and hand," *J. Biomech.,* vol. 38, no. 5, pp. 981-992, 2005.
[127] I. W. Charlton, P. Tate, P. Smyth, and L. Roren, "Repeatability of an optimised lower body model," *Gait & posture,* vol. 20, no. 2, pp. 213-221, 2004.
[128] V. Camomilla, A. Cereatti, G. Vannozzi, and A. Cappozzo, "An optimized protocol for hip joint centre determination using the functional method," *Journal of biomechanics,* vol. 39, no. 6, pp. 1096-1106, 2006.
[129] A. Ferrari *et al.*, "Quantitative comparison of five current protocols in gait analysis," *Gait & posture,* vol. 28, no. 2, pp. 207-216, 2008.
[130] A. Leardini, Z. Sawacha, G. Paolini, S. Ingrosso, R. Nativo, and M. G. Benedetti, "A new anatomically based protocol for gait analysis in children," *Gait & posture,* vol. 26, no. 4, pp. 560-571, 2007.
[131] M. Donati, V. Camomilla, G. Vannozzi, and A. Cappozzo, "Anatomical frame identification and reconstruction for repeatable lower limb joint kinematics estimates," *Journal of biomechanics,* vol. 41, no. 10, pp. 2219-2226, 2008.
[132] T. D. Collins, S. N. Ghoussayni, D. J. Ewins, and J. A. Kent, "A six degrees-of-freedom marker set for gait analysis: repeatability and comparison with a modified Helen Hayes set," *Gait & posture,* vol. 30, no. 2, pp. 173-180, 2009.
[133] A. Peters, M. Sangeux, M. E. Morris, and R. Baker, "Determination of the optimal locations of surface-mounted markers on the tibial segment," *Gait & Posture,* vol. 29, no. 1, pp. 42-48, 2009.
[134] M. Żuk and C. Pezowicz, "Kinematic analysis of a six-degrees-of-freedom model based on ISB recommendation: a repeatability analysis and comparison with conventional gait model," *Applied bionics and biomechanics,* vol. 2015, 2015.
[135] M. Manca *et al.*, "Repeatability of a new protocol for gait analysis in adult subjects," *Gait & posture,* vol. 32, no. 2, pp. 282-284, 2010.
[136] F. Stief, H. Böhm, K. Michel, A. Schwirtz, and L. Döderlein, "Reliability and accuracy in three-dimensional gait analysis: a comparison of two lower body protocols," *Journal of applied biomechanics,* vol. 29, no. 1, pp. 105-111, 2013.
[137] C. McGibbon, J. Fowler, S. Chase, K. Steeves, J. Landry, and A. Mohamed, "Evaluation of anatomical and functional hip joint center methods: the effects of activity type, gender, and proximal reference segment," *Journal of biomechanical engineering,* vol. 138, no. 1, 2016.
[138] H. Kainz *et al.*, "Reliability of four models for clinical gait analysis," *Gait & posture,* vol. 54, pp. 325-331, 2017.

[139] J. M. Wilken, K. M. Rodriguez, M. Brawner, and B. J. Darter, "Reliability and minimal detectible change values for gait kinematics and kinetics in healthy adults," *Gait & posture,* vol. 35, no. 2, pp. 301-307, 2012.
[140] E. Szczerbik and M. Kalinowska, "The influence of knee marker placement error on evaluation of gait kinematic parameters," *Acta Bioeng. Biomech,* vol. 13, no. 3, pp. 43-46, 2011.
[141] D. Meldrum, C. Shouldice, R. Conroy, K. Jones, and M. Forward, "Test–retest reliability of three dimensional gait analysis: including a novel approach to visualising agreement of gait cycle waveforms with Bland and Altman plots," *Gait & posture,* vol. 39, no. 1, pp. 265-271, 2014.
[142] K. Chia and M. Sangeux, "Quantifying sources of variability in gait analysis," *Gait & posture,* vol. 56, pp. 68-75, 2017.
[143] F. C. Anderson and M. G. Pandy, "Static and dynamic optimization solutions for gait are practically equivalent," *Journal of biomechanics,* vol. 34, no. 2, pp. 153-161, 2001.
[144] S. L. Delp *et al.*, "OpenSim: open-source software to create and analyze dynamic simulations of movement," *IEEE transactions on biomedical engineering,* vol. 54, no. 11, pp. 1940-1950, 2007.
[145] A. Seth *et al.*, "OpenSim: Simulating musculoskeletal dynamics and neuromuscular control to study human and animal movement," *PLoS computational biology,* vol. 14, no. 7, p. e1006223, 2018.
[146] D. G. Thelen and F. C. Anderson, "Using computed muscle control to generate forward dynamic simulations of human walking from experimental data," *Journal of biomechanics,* vol. 39, no. 6, pp. 1107-1115, 2006.
[147] D. G. Thelen, F. C. Anderson, and S. L. Delp, "Generating dynamic simulations of movement using computed muscle control," *Journal of biomechanics,* vol. 36, no. 3, pp. 321-328, 2003.
[148] M. A. Sherman, A. Seth, and S. L. Delp, "Simbody: multibody dynamics for biomedical research," *Procedia Iutam,* vol. 2, pp. 241-261, 2011.
[149] J. L. Hicks, T. K. Uchida, A. Seth, A. Rajagopal, and S. L. Delp, "Is my model good enough? Best practices for verification and validation of musculoskeletal models and simulations of movement," *Journal of biomechanical engineering,* vol. 137, no. 2, 2015.
[150] R. Baker, "Gait analysis methods in rehabilitation," *Journal of neuroengineering and rehabilitation,* vol. 3, no. 1, pp. 1-10, 2006.
[151] S. M. Kidder, F. S. Abuzzahab, G. F. Harris, and J. E. Johnson, "A system for the analysis of foot and ankle kinematics during gait," *IEEE transactions on rehabilitation engineering,* vol. 4, no. 1, pp. 25-32, 1996.
[152] Y. Zhang, K. Chen, J. Yi, T. Liu, and Q. Pan, "Whole-body pose estimation in human bicycle riding using a small set of wearable sensors," *IEEE/ASME Transactions on Mechatronics,* vol. 21, no. 1, pp. 163-174, 2015.
[153] J.-L. Blanco, "A tutorial on se (3) transformation parameterizations and on-manifold optimization," *University of Malaga, Tech. Rep,* vol. 3, 2010.
[154] M. Pharr, W. Jakob, and G. Humphreys, *Physically based rendering: From theory to implementation*. Morgan Kaufmann, 2016
[155] Y.-B. Jia, "Quaternions and rotations," *Com S,* vol. 477, no. 577, p. 15, 2008.
[156] M. Pedley, "Tilt sensing using a three-axis accelerometer," *Freescale semiconductor application note,* vol. 1, pp. 2012-2013, 2013.
[157] H. Zhou and H. Hu, "Human motion tracking for rehabilitation—A survey," *Biomedical signal processing and control,* vol. 3, no. 1, pp. 1-18, 2008.
[158] R. E. Kalman, "A new approach to linear filtering and prediction problems," 1960.
[159] N. Thacker and A. Lacey, "Tutorial: The kalman filter," *Imaging Science and Biomedical Engineering Division, Medical School, University of Manchester,* p. 61, 1998.
[160] Y. Kim and H. Bang, "Introduction to Kalman filter and its applications," in *Introduction and Implementations of the Kalman Filter*: IntechOpen, 2018.
[161] M. Kok, J. D. Hol, and T. B. Schön, "Indoor positioning using ultrawideband and inertial measurements," *IEEE Transactions on Vehicular Technology,* vol. 64, no. 4, pp. 1293-1303, 2015.

[162] M. Trumble, A. Gilbert, C. Malleson, A. Hilton, and J. P. Collomosse, "Total Capture: 3D Human Pose Estimation Fusing Video and Inertial Sensors," in *BMVC*, 2017, vol. 2, no. 5, pp. 1-13
[163] M. B. Alatise and G. P. Hancke, "Pose estimation of a mobile robot based on fusion of IMU data and vision data using an extended Kalman filter," *Sensors,* vol. 17, no. 10, p. 2164, 2017.
[164] A. Tobergte, M. Pomarlan, and G. Hirzinger, "Robust multi sensor pose estimation for medical applications," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2009, pp. 492-497: IEEE
[165] C. A. Hammond, G. L. Hatfield, M. K. Gilbart, S. J. Garland, and M. A. Hunt, "Trunk and lower limb biomechanics during stair climbing in people with and without symptomatic femoroacetabular impingement," *Clinical biomechanics,* vol. 42, pp. 108-114, 2017.
[166] M. A. Hunt, J. R. Gunether, and M. K. Gilbart, "Kinematic and kinetic differences during walking in patients with and without symptomatic femoroacetabular impingement," *Clinical biomechanics,* vol. 28, no. 5, pp. 519-523, 2013.
[167] S. Sabharwal and A. Kumar, "Methods for assessing leg length discrepancy," *Clin. Orthop. Relat. Res.,* vol. 466, no. 12, pp. 2910-2922, 2008.
[168] A. Prado, X. Cao, X. Ding, and S. Agrawal, "Prediction of Gait Cycle Percentage Using Instrumented Shoes with Artificial Neural Networks," in *2020 IEEE International Conference on Robotics and Automation*, 2020: IEEE
[169] Z. Wang and T. Oates, "Imaging time-series to improve classification and imputation," *arXiv preprint arXiv:1506.00327,* 2015.
[170] F. Cremer, M. Urbazaev, C. Berger, M. D. Mahecha, C. Schmullius, and C. Thiel, "An image transform based on temporal decomposition," *IEEE Geoscience and Remote Sensing Letters,* vol. 15, no. 4, pp. 537-541, 2018.
[171] A. Vaswani *et al.*, "Attention is all you need," *arXiv preprint arXiv:1706.03762,* 2017.
[172] F. Calefato, F. Lanubile, and N. Novielli, "Emotxt: a toolkit for emotion recognition from text," in *2017 seventh international conference on Affective Computing and Intelligent Interaction Workshops and Demos (ACIIW)*, 2017, pp. 79-80: IEEE
[173] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," *arXiv preprint arXiv:1508.04025,* 2015.

APPENDIX

A: Eulers Angles Conventions

Table A.1: Eulers Angles Conventions

Proper Euler angles	Talt-Bryan (Cardon) angles
$X_1Z_2X_3 = \begin{bmatrix} c_2 & -c_3s_2 & s_2s_3 \\ c_1s_2 & c_1c_2c_3 - s_1s_3 & -c_3s_1 - c_1c_2s_3 \\ s_1s_2 & c_1s_3 + c_2c_3s_1 & c_1c_3 - c_2s_1s_3 \end{bmatrix}$	$X_1Z_2Y_3 = \begin{bmatrix} c_2c_3 & -s_2 & c_2s_3 \\ s_1s_3 + c_1c_3s_2 & c_1c_2 & c_1s_2s_3 - c_3s_1 \\ c_3s_1s_2 - c_1s_3 & c_2s_1 & c_1c_3 + s_1s_2s_3 \end{bmatrix}$
$X_1Y_2X_3 = \begin{bmatrix} c_2 & s_2s_3 & c_3s_2 \\ s_1s_2 & c_1c_3 - c_2s_1s_3 & -c_1s_3 - c_2c_3s_1 \\ -c_1s_2 & c_3s_1 + c_1c_2s_3 & c_1c_2c_3 - s_1s_3 \end{bmatrix}$	$X_1Y_2Z_3 = \begin{bmatrix} c_2c_3 & -c_2s_3 & s_2 \\ c_3s_1s_2 + c_1s_3 & c_1c_3 - s_1s_2s_3 & -c_2s_1 \\ s_1s_3 - c_1c_3s_2 & c_1s_2s_3 + c_3s_1 & c_1c_2 \end{bmatrix}$
$Y_1X_2Y_3 = \begin{bmatrix} c_1c_3 - c_2s_1s_3 & s_1s_2 & c_1s_3 + c_2c_3s_1 \\ s_2s_3 & c_2 & -c_3s_2 \\ -c_3s_1 - c_1c_2s_3 & c_1s_2 & c_1c_2c_3 - s_1s_3 \end{bmatrix}$	$Y_1X_2Z_3 = \begin{bmatrix} c_1c_3 + s_1s_2s_3 & c_3s_1s_2 - c_1s_3 & c_2s_1 \\ c_2s_3 & c_2c_3 & -s_2 \\ c_1s_2s_3 - c_3s_1 & s_1s_3 + c_1c_3s_2 & c_1c_2 \end{bmatrix}$
$Y_1Z_2Y_3 = \begin{bmatrix} c_1c_2c_3 - s_1s_3 & -c_1s_2 & c_3s_1 + c_1c_2s_3 \\ c_3s_2 & c_2 & s_2s_3 \\ -c_1s_3 - c_2c_3s_1 & s_1s_2 & c_1c_3 - c_2s_1s_3 \end{bmatrix}$	$Y_1Z_2X_3 = \begin{bmatrix} c_1c_2 & s_1s_3 - c_1c_3s_2 & c_1s_2s_3 + c_3s_1 \\ s_2 & c_2c_3 & -c_2s_3 \\ -c_2s_1 & c_3s_1s_2 + c_1s_3 & c_1c_3 - s_1s_2s_3 \end{bmatrix}$
$Z_1Y_2Z_3 = \begin{bmatrix} c_1c_2c_3 - s_1s_3 & -c_3s_1 - c_1c_2s_3 & c_1s_2 \\ c_1s_3 + c_2c_3s_1 & c_1c_3 - c_2s_1s_3 & s_1s_2 \\ -c_3s_2 & s_2s_3 & c_2 \end{bmatrix}$	$Z_1Y_2X_3 = \begin{bmatrix} c_1c_2 & c_1s_2s_3 - c_3s_1 & s_1s_3 + c_1c_3s_2 \\ c_2s_1 & c_1c_3 + s_1s_2s_3 & c_3s_1s_2 - c_1s_3 \\ -s_2 & c_2s_3 & c_2c_3 \end{bmatrix}$
$Z_1X_2Z_3 = \begin{bmatrix} c_1c_3 - c_2s_1s_3 & -c_1s_3 - c_2c_3s_1 & s_1s_2 \\ c_3s_1 + c_1c_2s_3 & c_1c_2c_3 - s_1s_3 & -c_1s_2 \\ s_2s_3 & c_3s_2 & c_2 \end{bmatrix}$	$Z_1X_2Y_3 = \begin{bmatrix} c_1c_3 - s_1s_2s_3 & -c_2s_1 & c_3s_1s_2 + c_1s_3 \\ c_1s_2s_3 + c_3s_1 & c_1c_2 & s_1s_3 - c_1c_3s_2 \\ -c_2s_3 & s_2 & c_2c_3 \end{bmatrix}$

c and s represent cosine and sine respectively; X, Y, Z represent the rotational matrix about the x, y, z axis of the original coordinate system (extrinsic rotation). 1, 2, 3 represent the order of the rotation angles.

Table A.2: Extrinsic and intrinsic rotation equalivent for Talt Bryan angles

Extrinisic rotations	Intrinsic rotations
$X_1Z_2Y_3$	$Y_1Z_2'X_3''$
$X_1Y_2Z_3$	$Z_1Y_2'X_3''$
$Y_1X_2Z_3$	$Z_1X_2'Y_3''$
$Y_1Z_2X_3$	$X_1Z_2'Y_3''$
$Z_1Y_2X_3$	$X_1Y_2'Z_3''$
$Z_1X_2Y_3$	$Y_1X_2'Z_3''$

Extrinsic rotations occur about the axes of the original coordinate system where it is motionless (fixed). Intrinsic rotations occur about the axes of the rotating corrdinate system which changes after each elemental rotation, these transformed axes are indicated by $'$. X, Y, Z represent the rotational matrix about the x, y, z axis. 1, 2, 3 represent the order of the rotations

B: Quaternion Rotation: Matrix Formulation

We know when two quaternions multiply

$$pq = [1 \quad \mathbf{i} \quad \mathbf{j} \quad \mathbf{k}] \begin{bmatrix} p_0q_0 - p_1q_1 - p_2q_2 - p_3q_3 \\ p_0q_1 + p_1q_0 + p_2q_3 - p_3q_2 \\ p_0q_2 - p_1q_3 + p_2q_0 + p_3q_1 \\ p_0q_3 + p_1q_2 - p_2q_1 + p_3q_0 \end{bmatrix}$$

and

$$q = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k}$$

$$\mathbf{v} = v_x\mathbf{i} + v_y\mathbf{j} + v_z\mathbf{k}$$

$$q^{-1} = q_0 - (q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k})$$

Solve,

$$\mathbf{v}' = q\mathbf{v}q^{-1}$$

Let $q' = \mathbf{v}q^{-1}$

$$q' = \begin{bmatrix} q_0' \\ q_1' \\ q_2' \\ q_3' \end{bmatrix}$$

Solve $q' = \mathbf{v}q^{-1}$

$$q' = \mathbf{v}q^{-1} = [1 \quad \mathbf{i} \quad \mathbf{j} \quad \mathbf{k}] \begin{bmatrix} 0 + v_xq_1 + v_yq_2 + v_zq_3 \\ 0 + v_xq_0 - v_yq_3 + v_zq_2 \\ 0 + v_xq_3 + v_yq_0 - v_zq_1 \\ 0 - v_xq_2 + v_yq_1 + v_zq_0 \end{bmatrix}$$

Then,

$$q\mathbf{v}q^{-1} = qq'$$

Solve qq',

$$qq' = [1 \quad \mathbf{i} \quad \mathbf{j} \quad \mathbf{k}] \begin{bmatrix} q_0q_0' - q_1q_1' - q_2q_2' - q_3q_3' \\ q_0q_1' + q_1q_0' + q_2q_3' - q_3q_2' \\ q_0q_2' - q_1q_3' + q_2q_0' + q_3q_1' \\ q_0q_3' + q_1q_2' - q_2q_1' + q_3q_0' \end{bmatrix}$$

Expand the real part,

$$q_0q_0' = q_0v_xq_1 + q_0v_yq_2 + q_0v_zq_3$$

$$q_1q_1' = q_1v_xq_0 - q_1v_yq_3 + q_1v_zq_2$$

$$q_2q_2' = q_2v_xq_3 + q_2v_yq_0 - q_2v_zq_1$$

$$q_3q_3' = -q_3v_xq_2 + q_3v_yq_1 + q_3v_zq_0$$

Then,

$$q_0q_0' - q_1q_1' - q_2q_2' - q_3q_3'$$

$$= q_0 v_x q_1 - q_1 v_x q_0 + q_0 v_y q_2 - q_2 v_y q_0 + q_0 v_z q_3 - q_3 v_z q_0 + q_1 v_y q_3 - q_3 v_y q_1 + (-q_1 v_z q_2)$$
$$+ q_2 v_z q_1 + (-q_2 v_x q_3) + q_3 v_x q_2$$
$$= 0$$

Expand the $\mathbf{i}$ part,

$$q_0 q_1' = q_0 v_x q_0 - q_0 v_y q_3 + q_0 v_z q_2$$
$$q_1 q_0' = q_1 v_x q_1 + q_1 v_y q_2 + q_1 v_z q_3$$
$$q_2 q_3' = -q_2 v_x q_2 + q_2 v_y q_1 + q_2 v_z q_0$$
$$q_3 q_2' = q_3 v_x q_3 + q_3 v_y q_0 - q_3 v_z q_1$$

Then,

$$q_0 q_1' + q_1 q_0' + q_2 q_3' - q_3 q_2'$$
$$= (q_0^2 + q_1^2 - q_2^2 - q_3^2) v_x + 2(q_1 q_2 - q_0 q_3) v_y + 2(q_1 q_3 + q_0 q_2) v_z$$

Expand the $\mathbf{j}$ part,

$$q_0 q_2' = q_0 v_x q_3 + q_0 v_y q_0 - q_0 v_z q_1$$
$$q_1 q_3' = -q_1 v_x q_2 + q_1 v_y q_1 + q_1 v_z q_0$$
$$q_2 q_0' = q_2 v_x q_1 + q_2 v_y q_2 + q_2 v_z q_3$$
$$q_3 q_1' = q_3 v_x q_0 - q_3 v_y q_3 + q_3 v_z q_2$$

Then,

$$q_0 q_2' - q_1 q_3' + q_2 q_0' + q_3 q_1'$$
$$= 2(q_1 q_2 + q_0 q_3) v_x + (q_0^2 - q_1^2 + q_2^2 - q_3^2) v_y + 2(q_2 q_3 - q_0 q_1) v_z$$

Expand the $\mathbf{k}$ part,

$$q_0 q_3' = -q_0 v_x q_2 + q_0 v_y q_1 + q_0 v_z q_0$$
$$q_1 q_2' = q_1 v_x q_3 + q_1 v_y q_0 - q_1 v_z q_1$$
$$q_2 q_1' = q_2 v_x q_0 - q_2 v_y q_3 + q_2 v_z q_2$$
$$q_3 q_0' = q_3 v_x q_1 + q_3 v_y q_2 + q_3 v_z q_3$$

Then,

$$q_0 q_3' + q_1 q_2' - q_2 q_1' + q_3 q_0'$$
$$= 2(q_1 q_3 - q_0 q_2) v_x + 2(q_2 q_3 + q_0 q_1) v_y + (q_0^2 - q_1^2 - q_2^2 + q_3^2) v_z$$

Let the matrix form be:

$$\mathbf{v}' = [\mathbf{R_q}]_{\mathbf{3x3}}\, \mathbf{v}$$

Thus,

$$[\mathbf{R_q}]_{\mathbf{3x3}} = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1 q_2 - q_0 q_3) & 2(q_1 q_3 + q_0 q_2) \\ 2(q_1 q_2 + q_0 q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_2 q_3 + q_0 q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix}$$

C: Accuracy of Supervised Learning Models

Table C.1: Accuracy of the trained model on the training set and testing set for stair ascent

Type			Training set				Testing set			
Output	Cost	Model	Misclassification cost	Prediction speed	Training time	Overall Accuracy	IC /stance Accuracy	FO /swing Accuracy	Steady-state Accuracy	Overall Accuracy
2	1	1	1822	830000	10.364	98.3	0.0025	0.0025	0.9998	0.9824
		2	1745	1000000	8.8611	98.4	0.0000	0.0000	1.0000	0.9826
		3	1741	1100000	8.566	98.4	0.0000	0.0000	1.0000	0.9826
		4	2181	850000	6.4929	98	0.0000	0.0025	0.9979	0.9805
		5	5768	1600000	7.7663	94.7	0.0631	0.6398	0.9487	0.9383
		7	11929	1500000	9.6765	89	0.5271	0.8098	0.8830	0.8792
		8	12109	200	2333.4	88.8	0.4678	0.7909	0.8911	0.8866
		9	1741	450000	105.96	98.4	0.0000	0.0000	1.0000	0.9826
		10	1741	29000	9826.6	98.4				
		11	23496	20000	14475	78.3				
		12	1741	11000	1551.6	98.4	0.0000	0.0000	1.0000	0.9826
		13	1741	17000	1727	98.4	0.0000	0.0000	1.0000	0.9826
		14	1741	20000	1835.4	98.4	0.0000	0.0000	1.0000	0.9826
		15	2868	77000	1843.8	97.4	0.0769	0.1222	0.9848	0.9694
		16	1806	32000	1858.2	98.3	0.0000	0.0164	0.9992	0.9819
		17	1741	9300	1904.6	98.4	0.0000	0.0000	1.0000	0.9826
		18	1780	710	2553.5	98.4	0.0000	0.0101	0.9994	0.9821
		19	1800	6100	2402.8	98.3	0.0000	0.0176	0.9992	0.9819
		20	1848	34000	2417.6	98.3	0.0050	0.0151	0.9988	0.9816
		21	1741	110000	2515	98.4	0.0000	0.0000	1.0000	0.9826
		22	1798	60000	2621.6	98.3	0.0013	0.0126	0.9993	0.9820
		23	1765	51000	2588.2	98.4	0.0000	0.0000	1.0000	0.9826
		24	2066	5800	2697.5	98.1	0.0088	0.0428	0.9975	0.9806

		25	24370	110000	2645.8	77.5	0.9042	0.9207	0.7703	0.7728
2	2	1	8648	1600000	7.6984	96.7	0.6108	0.7569	0.8562	0.8412
		2	8784	1600000	5.6181	97.6	0.5364	0.8370	0.9202	0.8998
		3	8705	1700000	4.9257	98.4	0.0000	0.0000	1.0000	0.9129
		4	10538	110000	6.8198	96.3	0.0000	0.0000	1.0000	0.9129
		5	15269	710000	8.2751	87.9	0.0000	0.0111	0.9556	0.8728
		7	20954	780000	9.1267	81.8	0.4905	0.6927	0.8735	0.8489
		8	18658	190	2442.5	84.1	0.0409	0.2531	0.9885	0.9153
		9	8705	440000	335.9	98.4	0.1644	0.2282	0.9737	0.9060
		10	11454	19000	11964	95.7	0.0694	0.0874	0.9892	0.9099
		12	8060	6900	956.52	97.3	0.0035	0.2000	0.9906	0.9133
		13	8705	7000	1905.1	98.4	0.0000	0.0000	1.0000	0.9129
		14	8705	7200	2291.2	98.4	0.0000	0.0000	1.0000	0.9129
		15	8680	83000	2299.6	97.4	0.0257	0.1224	0.9918	0.9119
		16	8658	34000	2314.9	95.7	0.0000	0.0000	1.0000	0.9129
		17	8063	9700	2359.2	97.4	0.0235	0.1184	0.9927	0.9124
		18	9443	690	3019	95.5	0.1508	0.2645	0.9741	0.9074
		19	8681	6000	2512.5	95.7	0.1682	0.2239	0.9740	0.9062
		20	8703	34000	2527.7	96	0.1463	0.2030	0.9767	0.9069
		21	8708	110000	2625.1	98.4	0.0000	0.0000	1.0000	0.9129
		22	8492	65000	2734.1	98.1	0.0111	0.0360	0.9978	0.9130
		23	8833	60000	2761.4	98.3	0.0000	0.0018	0.9998	0.9128
		24	8631	7800	2852.1	98.3	0.0005	0.0060	0.9997	0.9129
		25	24240	92000	2877.5	77.8	0.8711	0.8602	0.8162	0.8205
1	1	1	3918	710000	13.023	96.4	0.9580	0.9232		0.9427
		2	4310	670000	11.164	96	0.9648	0.9146		0.9428
		3	6430	860000	10.758	94.1	0.9533	0.8984		0.9292
		4	5596	550000	8.4744	94.8	0.9700	0.9015		0.9399
		5	5185	1300000	9.265	95.2	0.9610	0.9345		0.9493
		6		750000	23.33	95.8	0.9666	0.9246		0.9481
		7	6570	1300000	12.27	93.9	0.9601	0.9133		0.9395

		8	6324	170	2810.8	94.2	0.9561	0.9251		0.9425
		9	4516	19000	5624.3	95.8	0.9651	0.9281		0.9489
		10	4096	12000	19982	96.2	0.9647	0.9304		0.9496
		11	35096	11000	21273	67.7	0.7176	0.4382		0.5949
		12	3015	9500	3624.7	97.2	0.9616	0.9337		0.9493
		13	3518	8100	3992.8	96.8	0.9655	0.9345		0.9519
		14	4297	7000	4421.1	96	0.9664	0.9317		0.9511
		15	3767	50000	4432.7	96.5	0.9503	0.9230		0.9383
		16	3072	21000	4454.8	97.2	0.9565	0.9333		0.9463
		17	3358	6900	4517.6	96.9	0.9629	0.9295		0.9482
		18	3766	1100	4400	96.5	0.9576	0.9235		0.9426
		19	3080	3800	5668	97.2	0.9565	0.9325		0.9459
		20	3064	17000	5650.4	97.2	0.9589	0.9266		0.9447
		21	3845	67000	5797.7	96.5	0.9651	0.9223		0.9463
		22	3108	33000	5904.6	97.1	0.9581	0.9329		0.9470
		23	6824	37000	5843.4	93.7	0.9839	0.8544		0.9270
		24	3889	4100	6003.8	96.4	0.9635	0.9259		0.9470
		25	4119	81000	6065.5	96.2	0.9587	0.9250		0.9439
3	1	1	7837	770000	11.186	92.8	0.2598	0.3388	0.9213	0.9105
		2	8665	860000	10.184	92	0.0164	0.0000	0.9983	0.9811
		3	8705	920000	8.9182	92	0.0000	0.0000	1.0000	0.9826
		4	9351	740000	6.5412	91.4	0.1400	0.3577	0.9295	0.9177
		5	11192	1100000	8.5101	89.7	0.4842	0.8073	0.8516	0.8480
		7	18465	1200000	11.22	83	0.8008	0.8463	0.7749	0.7758
		8	16189	160	2856	85.1	0.6709	0.8778	0.8043	0.8037
		9	8165	610000	7700.4	92.5	0.0000	0.2733	0.9730	0.9584
		10	7033	11000	19263	93.5	0.0000	0.5542	0.9576	0.9457
		12	5716	4000	836.52	94.7	0.2018	0.4181	0.9363	0.9254
		13	7023	4200	1560.9	93.5	0.0000	0.5126	0.9554	0.9432
		14	8437	3300	2143.8	92.2	0.0000	0.1461	0.9833	0.9675
		15	6796	42000	2157.3	93.7	0.3001	0.4219	0.9168	0.9071

		16	5862	17000	2186.1	94.6	0.1904	0.4055	0.9304	0.9194
		17	6256	6300	2257.4	94.2	0.1652	0.3275	0.9386	0.9265
		18	6729	580	3080.4	93.8	0.1526	0.5088	0.9318	0.9213
		19	5895	4700	2945.8	94.6	0.1917	0.3992	0.9301	0.9190
		20	5749	24000	2966.6	94.7	0.2421	0.3778	0.9282	0.9174
		21	7765	76000	3102.1	92.8	0.0000	0.2657	0.9798	0.9650
		22	5766	35000	3270.7	94.7	0.1803	0.4370	0.9345	0.9237
		23	9189	38000	3145.9	91.5	0.0101	0.1285	0.9701	0.9545
		24	6745	3300	3319.2	93.8	0.1652	0.3438	0.9409	0.9289
		25	20976	78000	3314.8	80.7	0.9369	0.9194	0.7285	0.7320
3	2	1	15972	1000000	11.142	88.7	0.5831	0.8408	0.8742	0.8601
		2	19568	1200000	9.6188	85.8	0.8308	0.8917	0.8107	0.8151
		3	24095	1200000	15.744	80.4	0.8875	0.9196	0.7752	0.7864
		4	21503	580000	12.756	87.5	0.7032	0.8534	0.8635	0.8561
		5	19938	420000	16.028	83.8	0.8963	0.9335	0.8101	0.8192
		7	29767	600000	18.539	75.2	0.9032	0.9408	0.7174	0.7352
		8	24012	180	2580.6	80	0.8224	0.9380	0.7802	0.7889
		9	18840	480000	12611	88.6	0.7433	0.8725	0.8838	0.8772
		10	20351	12000	20615	89.6	0.7980	0.0662	0.9030	0.8620
		12	11337	3900	4064	92.3	0.6230	0.8320	0.9035	0.8882
		13	13756	3300	5028.2	90.1	0.7395	0.8897	0.8843	0.8783
		14	17452	2300	6235.4	88.4	0.7927	0.9121	0.8825	0.8799
		15	20716	53000	6245.7	93.7	0.3231	0.5264	0.9441	0.8989
		16	12177	21000	6267.4	91.4	0.6368	0.8156	0.8861	0.8722
		17	13626	6900	6328	90.1	0.6610	0.8668	0.8813	0.8711
		18	14640	540	7196.6	90.2	0.6820	0.8592	0.8736	0.8646
		19	12148	4800	7283.9	91.4	0.6348	0.8191	0.8857	0.8719
		20	12091	22000	7305.1	92.1	0.6045	0.7937	0.8929	0.8761
		21	17739	78000	7431.4	86.2	0.8116	0.9023	0.8375	0.8392
		22	16064	33000	7634.8	94.2	0.3253	0.6431	0.9446	0.9045
		23	28411	46000	7670.2	87.8	0.5546	0.7438	0.8885	0.8677

24	29837	5700	7790.4	93.3	0.1032	0.3834	0.9773	0.9134
25	21535	79000	7830	80.8	0.9145	0.9453	0.7765	0.7898

IC: initial contact; FO: foot off. Output type 1 is stance and swing output; output type 2 and 3 are transition outputs with IC, FO and steady state.

Table C.2: Accuracy of the trained model on the training set and testing set for stair descent

Type			Training set				Testing set			
Output	Cost	Model	Misclassification cost	Prediction speed	Training time	Overall Accuracy	IC /stance Accuracy	FO /swing Accuracy	Steady-state Accuracy	Overall Accuracy
2	1	1	1823	840000	1789.9	98.2	0.0023	0.0057	0.9991	0.9800
		2	1761	810000	1788.9	98.3	0.0000	0.0000	1.0000	0.9808
		3	1745	880000	1788.5	98.3	0.0000	0.0000	1.0000	0.9808
		4	1995	560000	1785.2	98.1	0.0000	0.0011	1.0000	0.9808
		5	5860	1200000	1784.1	94.3	0.0978	0.7693	0.9435	0.9337
		7	16780	930000	1759.6	83.8	0.5324	0.9636	0.8385	0.8368
		8	16227	150	4196.2	84.3	0.5472	0.9636	0.8347	0.8332
		9	1745	310000	1896.2	98.3	0.0000	0.0000	1.0000	0.9808
		10	1745	7500	15992	98.3	0.0000	0.0000	1.0000	0.9808
		11	21594	5900	25750	79.2	0.0000	0.3182	0.9487	0.9335
		12	1745	5400	3543	98.3	0.0000	0.0000	1.0000	0.9808
		13	1745	7300	3816.9	98.3	0.0000	0.0000	1.0000	0.9808
		14	1745	9500	3993.7	98.3	0.0000	0.0000	1.0000	0.9808
		15	2927	64000	4012.9	97.2	0.0887	0.1125	0.9822	0.9653
		16	1818	24000	4031.5	98.2	0.0000	0.0182	0.9991	0.9801
		17	1745	7400	4088.4	98.3	0.0000	0.0000	1.0000	0.9808
		18	1772	780	4645.9	98.3	0.0023	0.0114	0.9991	0.9801
		19	1794	4200	4301.8	98.3	0.0000	0.0193	0.9992	0.9802
		20	1834	22000	4323.6	98.2	0.0057	0.0159	0.9984	0.9794
		21	1745	68000	4451.3	98.3	0.0000	0.0000	1.0000	0.9808
		22	1766	43000	4592.7	98.3	0.0011	0.0193	0.9989	0.9800
		23	1792	36000	4630.5	98.3	0.0000	0.0000	1.0000	0.9808
		24	2042	4500	4757.6	98	0.0148	0.0477	0.9953	0.9768
		25	25608	87000	4680.3	75.3	0.8123	0.9920	0.7624	0.7651
2	2	1	8735	860000	13.079	96.6	0.0478	0.1616	0.9931	0.9079
		2	8625	930000	11.909	98	0.0000	0.0907	0.9979	0.9066
		3	8725	940000	11.184	98.3	0.0000	0.0000	1.0000	0.9041

		4	9344	680000	8.4801	97.2	0.0014	0.1468	0.9855	0.8981
		5	14306	1200000	11.322	87.9	0.4391	0.7180	0.9018	0.8708
		7	22495	1000000	814.26	79	0.6710	0.9509	0.8474	0.8439
		8	21417	140	3160.1	80	0.6705	0.9332	0.8518	0.8470
		9	8725	150000	813.58	98.3	0.0000	0.0000	1.0000	0.9041
		10	9247	5700	14834	97.8	0.1970	0.2373	0.4873	0.4614
		11	29279	5400	25334	77.55	0.0018	0.2655	0.5064	0.4706
		12	8020	3500	1580.8	97.2	0.0710	0.1914	0.9888	0.9066
		13	8720	3000	2904	98.3	0.0000	0.0059	1.0000	0.9044
		14	8725	3000	3560.5	98.3	0.0000	0.0000	1.0000	0.9041
		15	8811	47000	3177.2	97.2	0.0669	0.1061	0.9878	0.9013
		16	8995	21000	3200.4	95.4	0.1759	0.2848	0.9720	0.9009
		17	8187	6700	3263.7	97.2	0.0592	0.1841	0.9898	0.9066
		18	9603	800	3837.5	95.3	0.1556	0.2784	0.9743	0.9017
		19	8946	4300	3688	95.4	0.1727	0.2866	0.9721	0.9009
		20	8961	23000	3710.3	95.8	0.1613	0.2530	0.9741	0.9005
		21	8662	90000	3851.6	98.2	0.0000	0.0834	0.9985	0.9067
		22	8473	46000	3982.6	98	0.0116	0.0327	0.9975	0.9039
		23	8952	51000	3881.3	97.9	0.0000	0.0277	0.9992	0.9047
		24	8535	6400	3985.7	98.3	0.0020	0.0116	0.9994	0.9042
		25	25728	110000	4007.3	75.3	0.8287	0.9252	0.8188	0.8244
1	1	1	4037	800000	17.106	96.1	0.9556	0.9308		0.9456
		2	4587	490000	12.882	85.6	0.9454	0.9437		0.9447
		3	5906	950000	12.359	94.3	0.9367	0.9576		0.9451
		4	10805	380000	9.9692	89.6	0.9769	0.7874		0.9010
		5	6906	1400000	9.3242	93.3	0.9307	0.9304		0.9306
		6		950000	818.54	92	0.9634	0.8654		0.9241
		7	6803	1000000	18.352	93.4	0.9032	0.9683		0.9293
		8	6710	140	3048.3	93.5	0.8870	0.9729		0.9214
		9	8186	3500	8176.3	92.1	0.9718	0.8545		0.9248
		10	3895	7600	12412	96.2	0.9659	0.9476		0.9585
		12	2997	3100	3897.6	97.1	0.9619	0.9412		0.9536

		13	3710	2600	4496.5	96.4	0.9571	0.9496		0.9541
		14	4614	2000	5218.9	95.5	0.9546	0.9421		0.9496
		15	3889	56000	5241.9	96.2	0.9458	0.9232		0.9367
		16	3334	23000	5261.3	96.8	0.9460	0.9415		0.9442
		17	3692	7100	5323.1	96.4	0.9484	0.9450		0.9470
		18	4378	680	5959.4	95.8	0.9505	0.9406		0.9465
		19	3330	4200	6059.9	96.8	0.9462	0.9416		0.9444
		20	3210	24000	6080.2	96.9	0.9510	0.9325		0.9436
		21	4057	77000	6200.3	96.1	0.9512	0.9442		0.9484
		22	3099	35000	6370.4	97	0.9519	0.9408		0.9474
		23	11331	46000	6402.9	89.1	0.9850	0.7769		0.9016
		24	3788	3400	6609.1	96.3	0.9417	0.9375		0.9400
		25	4473	81000	6685.3	95.7	0.9432	0.9532		0.9472
3	1	1	7089	640000	16.562	93.2	0.1832	0.4795	0.9259	0.9145
		2	7615	910000	15.155	92.7	0.0000	0.5614	0.9449	0.9322
		3	8738	990000	13.897	91.6	0.0000	0.0000	1.0000	0.9808
		4	9094	380000	11.734	91.2	0.0114	0.4057	0.9742	0.9595
		5	10379	720000	13.995	90	0.4881	0.9648	0.8517	0.8493
		7	18194	1100000	16.397	82.4	0.6143	0.9920	0.7835	0.7839
		8	17368	120	3686.6	83.2	0.6439	0.9875	0.7914	0.7919
		9	8529	280000	6333.3	91.8	0.0000	0.0761	0.9923	0.9740
		10	7119	5600	22296	93.1	0.2503	0.2568	0.9294	0.9164
		11	44342	4600	28435	57.2	0.7258	0.7852	0.7940	0.7933
		12	5404	1600	5061.2	94.8	0.2514	0.4034	0.9305	0.9190
		13	6626	1800	5908.3	93.6	0.1422	0.4932	0.9348	0.9230
		14	8165	1700	6838.6	92.1	0.0000	0.2716	0.9731	0.9571
		15	6544	59000	6358.9	93.7	0.3060	0.4102	0.9152	0.9045
		16	5668	24000	6376.9	94.5	0.1980	0.3636	0.9374	0.9248
		17	6160	8000	6430.8	94.1	0.1729	0.4273	0.9397	0.9274
		18	6797	770	7011.4	93.4	0.1945	0.2909	0.9482	0.9347
		19	5681	4400	6937.2	94.5	0.1991	0.3693	0.9374	0.9249
		20	5563	29000	6955.2	94.6	0.2491	0.4011	0.9272	0.9157

		21	7445	96000	7077.3	92.8	0.0000	0.4932	0.9534	0.9398
		22	5438	42000	7173.7	94.8	0.2400	0.4420	0.9257	0.9145
		23	9053	48000	7104.7	91.3	0.0000	0.0500	0.9968	0.9781
		24	6916	6300	7208.8	93.3	0.1104	0.3955	0.9477	0.9344
		25	20156	97000	7211.3	80.5	0.7270	0.9852	0.7584	0.7602
3	2	1	15879	880000	16.693	88.8	0.7167	0.8882	0.8833	0.8756
		2	18685	840000	15.766	87.5	0.7024	0.8841	0.8735	0.8658
		3	21842	790000	14.898	83.5	0.7142	0.9259	0.8561	0.8526
		4	29177	520000	12.898	85.1	0.5181	0.9145	0.8526	0.8396
		5	18067	520000	15.114	85.3	0.8476	0.9298	0.8465	0.8505
		7	24116	740000	17.518	78.5	0.7968	0.9864	0.7910	0.8007
		8	22393	130	4124.3	79.9	0.8168	0.9800	0.8097	0.8182
		9	20020	87000	9245.6	85.8	0.7531	0.9391	0.8352	0.8363
		10	31414	4700	22819	79.4	0.8164	0.7795	0.8966	0.8871
		12	10568	1700	5658.6	92.3	0.7076	0.8445	0.9078	0.8952
		13	13270	1200	7046.6	90.3	0.7782	0.9007	0.8950	0.8896
		14	17003	980	8726	87.2	0.7656	0.9559	0.8697	0.8688
		15	19944	62000	8768.1	93.7	0.3879	0.5132	0.9482	0.9004
		16	11866	21000	8785.1	91.4	0.6621	0.8495	0.8946	0.8813
		17	12938	6800	8855.5	90.4	0.6646	0.9032	0.8936	0.8831
		18	15714	720	9487.7	89.2	0.6735	0.8036	0.8920	0.8772
		19	11884	4400	9381.7	91.3	0.6651	0.8459	0.8945	0.8811
		20	11602	22000	9405.6	92.2	0.6382	0.8239	0.9018	0.8854
		21	16635	96000	9560.4	88.3	0.7058	0.9145	0.8835	0.8765
		22	14335	40000	9705	94.5	0.4562	0.6502	0.9441	0.9066
		23	29777	46000	9588.6	83.7	0.3504	0.9766	0.8518	0.8338
		24	22587	4100	9728	93.8	0.2817	0.5486	0.9591	0.9069
		25	21192	97000	9743.9	80.4	0.8246	0.9745	0.8152	0.8233

IC: initial contact; FO: foot off. Output type 1 is stance and swing output; output type 2 and 3 are transition outputs with IC, FO and steady state.