Coordination of Vision and Body Movements For Prosthetic Control

Vijeth Rai

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington

2020

Reading Committee:

Eric Rombokas, Chair

Dr. Kat M. Steele

Dr. Patrick Aubin

Program Authorized to Offer Degree:
Electrical and Computer Engineering

University of Washington

Abstract

Coordination of Vision and Body Movements
For Prosthetic Control

Vijeth Rai

Chair of the Supervisory Committee:
Dr. Eric Rombokas
Mechanical Engineering

Powered lower limbs are getting more capable in their hardware capacity but their control is still challenging. The state-of-the-art simplifies the control needs for the vast multitude of real-world activities by categorizing them into a handful of commonly encountered activity classes or "modes", such as flatground mode, stair ascent mode, etc.

This dissertation aims to improve prosthetic lower limb control by focusing on the two challenges with using modes: 1) Estimating and handling the transitions between modes, and 2) Providing a larger repertoire of movements encompassing atypical and unstructured activities, such as side-shuffling and obstacle avoidance.

Human locomotion exploits coordination between vision and body movements to efficiently navigate an environment. It is a continuous motion fluidly adapting to the environment and not always categorizable into modes. We draw our inspiration from these facets of human movement to address the two challenges. We introduce a novel Coordinated Movement (CM) controller capable of generating continuous movements for all the desired movements. We use vision to directly sense the environment and anticipate transitions in advance.

Our CM controller exploits the strong inter-joint coordination exhibited in a typical movement to predict the trajectory of the prosthetic joint from the motion of the rest of the body. This novel approach *unifies* all the desired movements into a single controller and generates *continuous* kinematic reference trajectories without explicit modes or transitions. The underlying deep-learning model can be easily re-trained with new movement data to facilitate expansion of the movement vocabulary. Our real-time tests of the CM controller sheds light on practical challenges encountered when moving from offline analysis to real-time hardware, while also providing avenues for future improvements.

Vision sensors can be a window into the upcoming activities of the user to improve prosthetic transition performance. However, a significant bottleneck in employing vision classifiers to predict prosthetic mode labels is the resource-intensive process of manually labeling the data. This process is prone to subjective bias and limits the number of movements

to a handful of typical modes. We introduce an unsupervised method to label training data which allows the natural movements to dictate the number and characteristics of the generated modes. We demonstrate that a neural network model trained on these auto-generated mode labels can predict terrain changes prior to the kinematic changes of the user. Higher accuracy on a limited training dataset, and better generalizability is achieved by leveraging the technique of transfer learning.

The sensing and control strategies delineated in this dissertation can be applied towards a more natural experience of powered lower limbs.

TABLE OF CONTENTS

LIST OF FIGURES

DEDICATION

To Mom, Dad, Bava, and Akka dearest,
To Friends, too many to list, but here I'll try my best.

To Princi and Lentz, the mountain goats that run wild and free,
To the two gabby Gables, and Meyers - the fiery Missoula osprey,
And Pubi, Parker and the colorful Rangeeli.

To Fancy Nancy, Squirrel and Patty Poo,
Without yee, I'd be a cow that cant moo.

To Mendonsa and Mendoza, one with an 'S' and one with a 'Z',
An Alice who nurtures with her garden of love, and a Franz who holds his pee.

To Mrinalini and the Bay Area Bedis, the pillars of mine,
To Fanny and the French crew for the cheese and all that wine.

To rambunctious Rombo and a girl named Sie,
To the Schoendorf of Hadria, for the care and all the glee.

Some call it yogurt, some call it curds,
To Hegde, Boe, and Caballero for helping me find the right words.

To Dino, Smelly, Kutappa and the man who was a Moose,
To Bagha, Lindsay, and Ferleger for the lambs, ducks but not the goose,
And the drunken court of Lady Duchess, may yee never run out of booze.

To life, who whispers sweet nothings while she gently stabs you in the chest,
But for all her gifts, love, and wisdom, how could one not feel blessed.

And finally, to the one who endured and persisted
These words I say to a version of myself of the past -
The fruits of your labor are here, uncertain how long will it last,
But fear not the future, my friend, for the die is cast.

Chapter 1

INTRODUCTION

Powered limbs are getting more and more capable in their hardware capacity. The new devices are lighter and more powerful, with multiple joints and degrees of freedom [9, 25, 22]. The active power provided by powered prostheses mitigate compensatory gait patterns towards long-term benefits for users [104, 34, 53].

Given the vast space of possible movements, intuitive control of powered limbs however is still challenging [96]. **Categorization of activities into a handful of "locomotion modes" comprising the most commonly encountered terrains and activities such as flatground walking, stair ascent, etc. has been an effective and prominent strategy** [32, 41, 99]. The gait cycle within a mode is further divided into phases (swing, stance), each with distinct controllers and control parameters. The selection of mode for any given instant is done using pattern recognition of sensor data by machine learning models (SVM, LDA, or ANN, etc). The current state-of-the-art uses a combination of EMG and mechanical sensors as inputs to understand user intent *and* environment needs[91].

1.1 Motivation

Problem 1: Environment Transitions Successful prosthesis control has two important requirements: 1) Understanding user intent, 2) Understanding the dynamic needs of the environment and activity. Electro-myogram (EMG) and mechanical sensors are ideally suited for the first requirement as they provide a direct window into the user state. Their ability to understand the environmental features however, is limited and indirectly derived through user actions. They are also limited in their ability to provide a long window into the future actions of the user and the upcoming environment changes (Fig6.1). These factors have resulted in relatively worse performance during transitions. [31, 99].

EMG is also limited in its ability to recover from errors as the system is relying solely on the user state to indirectly gauge the demands of the current environment[32]. **Explicit sensing of the environment with ability to anticipate changes will improve robustness and overall performance** [96].

Problem 2: Unique and Non-Rhythmic Movements The mode based control strategy is efficient to tackle the most commonly encountered activities involving rhythmic movements. However, physically active powered prosthesis users cite challenges during unstruc-

tured activities, that require unique non-rhythmic movements (sports, getting in and out of cars)[28]. Users often prefer passive devices for their simplicity while adopting compensatory movements which can be physically demanding and detrimental [49, 50]. A possible solution could be to define more modes to encompass more types of activities. This requires the construction of more mode-specific controllers and makes the task of choosing the 'right' mode more difficult. A control strategy to address unstructured and non-rhythmic activities is necessary to fully leverage the benefits of powered limbs.

1.2 Research Aims

To summarize, the two limiting aspects of current mode-based prosthetic control are 1) Estimating and handling the transitions between modes, and 2) Providing a larger repertoire of movements. In this dissertation, we present three approaches that address these two problems.

Aim1: Visual Localization for Explicit Environment Sensing

The ability to anticipate transitions in advance would allow for triggering a change in controls at the right moment. EMG activity precedes a change in physical behavior with less than 100ms lead time [8]. For these reasons, vision as a sensing modality has recently garnered interest in research settings[112, 114, 107, 117, 52].

Our first approach addresses Problem 1 by using vision for explicit environment sensing. This method utilizes visual features to estimate the position of the prosthetic device in a pre-mapped environment. The appropriate mode for a given instant can be deduced from the estimated location. This requires prior knowledge of the map and the appropriate locomotion modes for the locations within it.

This approach improves fault tolerance by sensing the environment. The location of the prosthesis cross-referenced with known locations of transitions could be applied to anticipate upcoming transitions before the EMG or the body kinematics reflect the change.

Aim 2: Coordinated Movement: Continuous and Unified Control for Unstructured Terrains

Our second approach illustrates the data-driven CM controller which generates a *continuous* prosthetic reference trajectory, eliminating the need for the discretization of activities into modes and explicit transitions from one mode to another (Fig 4.5). The CM controller is *continuous* because it uses no concept of discrete gait events or phase, and *unified* because it is a single controller that provides control for multiple activities, without switching modes. Previous attempts [84, 78] at continuous prosthetic control have aimed at unifying the gait phases (swing, stance, etc) but within the same mode (flatground). The CM control strategy achieves continuity across gait phases and different gait modes.

We exploit the strong inter-joint coordination of human movement to predict the kinematic trajectory of a single joint based on the movement of the rest of the body. This

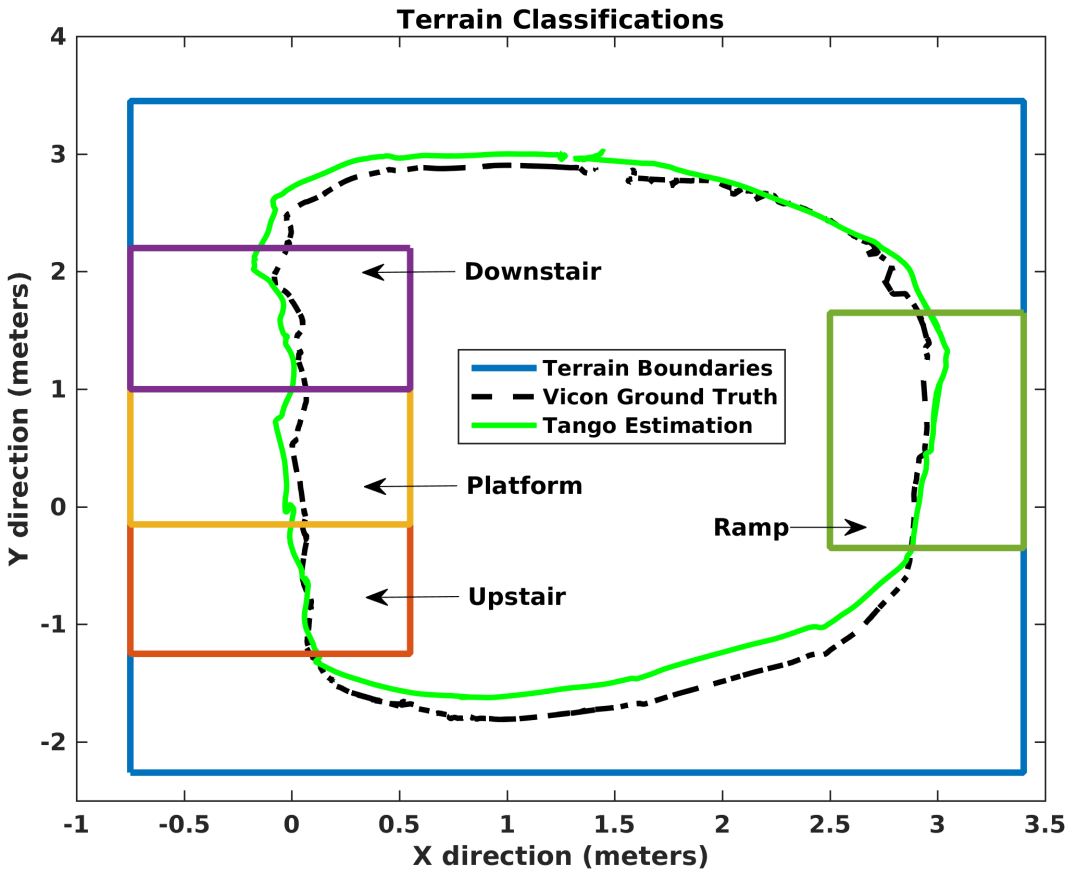


Figure 1.1: Example location data for a single trial from the VICON motion capture system and the mobile localization device.

predicted kinematic trajectory could be used as a reference to control a prosthetic actuator joint in real-time. All predictions had an Root Mean Squared Error (RMSE) of less than 7 degrees. .

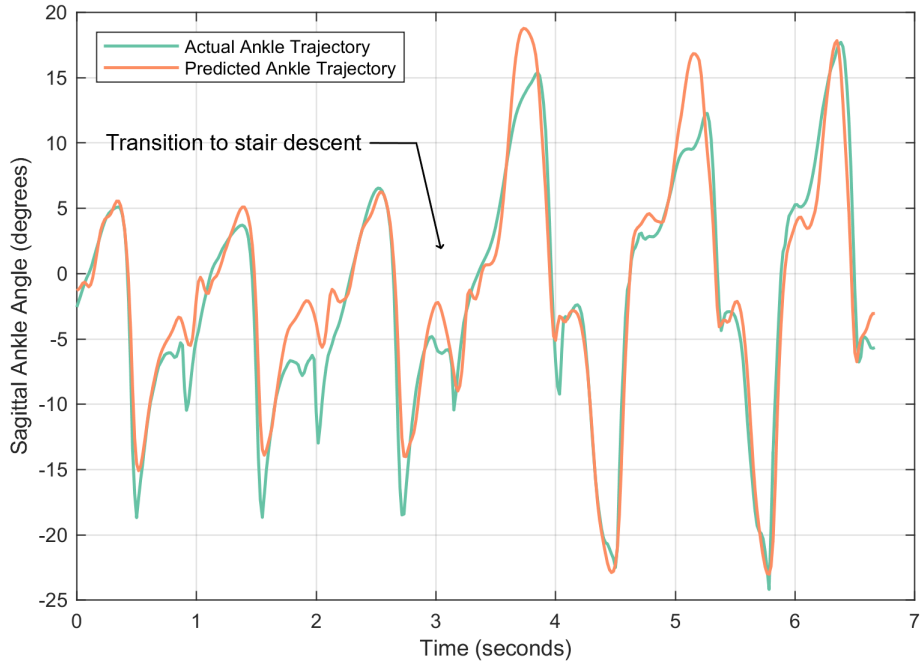


Figure 1.2: figure

Ankle joint predictions showing continuous seamless transition from flat ground walking to stair descent.

Aim 3: Unsupervised Labeling of Terrain Modes and Vision for Prosthesis Control

In our third aim, we address the need for a richer repertoire of movements while still retaining the benefits and tractability of a mode-based strategy (Problem 2). Mode-based control has the advantage of being more predictable and tractable, at least when the desired behaviors lie within the modes. However, optimizing mode-specific controllers can be laborious and is prone to bias. Here, we introduce a method for choosing modes that arise from the data, and for estimating them from vision.

We let the regularities present in natural movements dictate the number and characteristics of modes. Kinematically similar knee gait cycles are clustered. These cluster labels constitute "modes," by virtue of their similarity in the data, instead of human assumptions. This allows for *supervised* training of a vision classifier, using cluster labels as classes. We

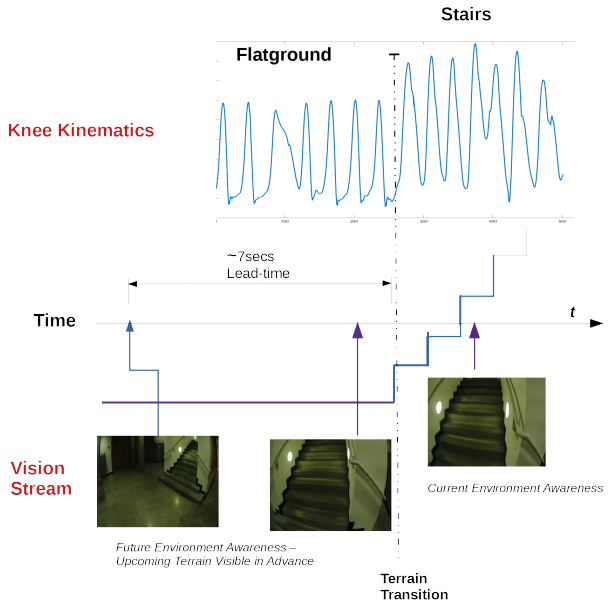


Figure 1.3: Vision sensor can foresee a new terrain before body kinematics change. In this example, the stair ascent was in the visual data stream almost 7 seconds prior to navigating the stairs. Vision can also provide information about current environment to provide fault tolerance and improve robustness

address Problem 1 by using the model generated labels to train the vision classifier to detect the environment and transitions almost 1.8 seconds prior.

Secondary Aim: Real-time Tests on Open Source Prosthetic Leg

We successfully evaluate the CM controller on a subject performing treadmill walking wearing a robotic prosthetic leg. This continues the work from Aim 2 which offered general insights to the approach using offline results. Actual real-time performance generally varies due to practical and physical constraints underscoring the need to evaluate performance on a real hardware testbed.

A CM controller was built to 1) acquire and process live kinematics as inputs from a wearable motion capture suit, 2) predict the missing joint kinematics (right knee) using a pre-trained neural network, and 3) actuate the knee joint of the Open Source Leg (OSL)[10] in real-time. The neural network was pre-trained on offline flat-ground walking data (to match the treadmill walking). The controller ran on a GPU powered laptop tethered to the OSL.

Several key insights and future improvements were highlighted during these tests. While the offline analysis in Aim 2 had the benefit of using the clean post-processed data, the real-time control did not. This, compounded by a significantly different calibration process compared to able-bodied subjects, impacted performance. The trade-off between the faster response time of the actuator and stiffer control was also observed and experiments on a bench-top platform were conducted to further explore this.

1.3 Dissertation Summary and Outline

This research aims to improve prosthetic control by focusing on performance upgrades while transitioning from one type of mode to another (Problem 1), and the movement demands of unstructured activities (Problem 2).

We present a novel continuous controller that obviates the categorization of movements into modes and an explicit transition mechanism. A single controller can effectively generate movements for all the desired activities. The control strategy was robust to subject-specific variations such as walking speed and step length. Prior to practical deployment, the choice of RMSE as a metric and the degree error reported in this study should be thoroughly substantiated. Experiments validating the suitability of prescribing normative kinematic trajectories collected from able-bodied subjects on amputee users are needed.

We present two methods for using computer vision to improve terrain understanding, especially for predicting transitions in advance. Despite high accuracy reported in studies, including this, the practical deployment will require rigorous testing in diverse and unstructured environments. A control scheme integrating vision with other sensor modalities and prosthesis should be the focus of future research.

Our real-time tests shed light on practical challenges encountered when moving from offline analysis to real-time hardware. In real-time, the lack of clean processed motion capture

data as inputs to the system can result in performance drop relative to offline predictions. Presumably, this could be mitigated by factoring in the noise in the training stage or by adapting an impedance-based control.

The goals and proceedings of this research are structured thus:

- Chapter 1: Introduction of motivation, research aims and implications of this research.
- Chapter 2: Background into current strategies for prosthesis control.
- Chapter 3: Aim 1 Explicit environment awareness using visual localization.
- Chapter 4: Aim 2 Data-driven continuous control for non-rhythmic movements without modes (Coordinated Movement (CM) Controller).
- Chapter 5 Aim 3 a) Automatic unsupervised labeling of training data to define new modes directly from data.
- Chapter 6 Aim 3 b) Vision classifier using auto-generated labels for explicit environment sensing and to anticipate transitions.
- Chapter 7: Real-time systems tests of the CM controller on a robotic prosthetic leg.
- Chapter 8: Conclusion.

Chapter 2

BACKGROUND

Background on prosthetic control

Most current prosthetic control architectures can generally be described as a 3-layer hierarchical Finite State Machine [96, 113]. Intent recognition or estimation of the user's locomotive intent is performed as the high level. The mid level translates this intent to generate a corresponding biomechanical reference trajectory. The low level comprises a device-specific controller responsible for tracking the trajectory.

At the high level, the controller must perceive the user's locomotive intent. Two main strategies are observed here are direct volitional and mode based control.

Direct Volitional Control Direct volitional control allows the user to directly modulate the actuator's position,torque based on EMG activity[38, 37]. Such functionality is suited for non-periodic or non-locomotive activities and those which require precise positioning of the limbs. Recently, [88] demonstrated bi-directional signalling using agonist-antagonist myoneural interface (AMI). AMI is capable of not just efferent control of an external prosthesis but also provide an amputee's central nervous system with musculotendinous proprioception. Direct volitional control is especially important in scenarios where the locomotive activity is irregular or noncyclic. However, most work in this is at a rudimentary phase and research has focused on cyclic and long-term repetitive activities in the form of mode based control.

Mode-based Control Most high-level intent recognition is based on modes of operation corresponding to activities, such as flat-ground walking, running, or stair ascent. Threshold-based rules have been used to trigger transitions from one mode to another in earlier devices[99]. More recently, machine learning techniques have used labelled examples of modes to estimate the current mode and transitions. Several different sensing modalities have been used for estimating the environment and the user intent. Mechanical sensors measuring forces, position, etc. are the most common. [99, 89, 61]. To get a more direct window into intent, there continues to be interest in using electromyogram (EMG) signals from the residual limb [41, 44]. Other sensors include socket capacitance[116]. We review these methodologies in the following sections.

2.1 Mode-based Control: Prior Art

2.1.1 Phase-based vs Non-phase-based control

In phase-based control, a set of actions is performed based on a programmed time delay following a clearly identifiable gait event. This control strategy relies heavily on the regularity of the gait phases. A weakness of this strategy is that it cannot handle abrupt mid-phase changes in activities. Each gait cycle must begin with the particular gait-phase (e.g, heel-strike). This strategy also inflexible to accommodate irregular or unprogrammed gait patterns (e.g, walking in a crowd or over rocky ground), unexpected events such as tripping. Most phase-based control are implemented as Finite State Machines decomposing individual gait phases, for each activity mode[86, 41]. This is the most popular strategy despite requiring several parameters to tune[87].

In non-phase based control, Complementary Limb Motion Estimation (CLME) infers the intended motion of affected limbs from the motion of the residual limbs, and maps this to a reference trajectory for robotic P/O joints to track[97]. This exploits the strong inter-joint coordination exhibited during physiological human motion[12]. The mapping is derived through regression of physiological gait recordings of healthy subjects.

A more recent success in non-phase based control strategy uses virtual constraints to define joint trajectories as functions of a monotonic phase variable that continuously represents the gait cycle across the entire stride[78]. The phase variable is computed from residual thigh motion, giving the amputee control over the timing of the prosthetic joint patterns. This strategy reduces the number of tuneable parameters. The holonomic nature of human gait, and the user-controlled progression of control allows performing activities such as backward walking and walking over obstacles. [84].

2.1.2 Sensing

Two basic requirements of safe, intuitive powered prosthesis control are: 1) Understanding user intent and, 2) Understanding environment and activity demands. For example, stair ascent or side shuffling into restaurant booth.

User-intent Sensing

Current control strategies focus more on user intent and indirectly gauge the environment by measuring user state. The sensing modalities used for user intent recognition are:

BCI and EEG: Despite a lot of recent interest in Brain Computer Interface (BCI) for neuro-prosthetics [64] and upper limb control, there have been limited studies using BCI in lower limb control. Only experiments with rhesus monkeys were found in the review[54]. Electroencephalogram (EEG) has garnered recent interest perhaps with advancement in the non-invasive sensor technologies [24].

Mechanical sensors: Some researchers [99] [89] [61] have applied multiple mechanical sensor feedback, such as 6-axis IMU, Force Sensitive Resistor and load cell for the purpose of volitional control. Initial results in [99] showed 100% identification of static locomotion mode and transition in and out of sitting to standing mode with a delay of 500ms. In comparative real-time studies, 84% of results achieved the desired identification [32]. However, authors in [18] state that a delay of more than 300 ms could be perceived adversely by the user and hence such a mechanical sensor based control would not be feasible. Despite their positives, mechanical sensors do not offer an easy window into the future intent of a user.

Electro-Myogram: EMG based volitional control has recently been studied in more detail [31][111][86][32] [110] following decades of research [18][74]. Huang et al showed that EMG and Pattern Recognition can be used to determine a user's current gait phase despite the non-stationary, time-varying features of EMG[41] with an average accuracy of 90% using a Linear Discriminant Analysis (LDA) classifier. Their results were more impressive with a fusion-based feature set using EMG and mechanical sensors with 99% accuracy in one study using a Suppor Vector Machine (SVM) classifier[42]; however, they were *post hoc* offline analysis. A subsequent case study by the same group showed Targeted Muscle Re-innervation (TMR) could further improve accuracy and reduce critical errors[31]. In a randomized clinical trial[32], the real-time accuracy for Neural Machine Interface (NMI) was approximately 92% with a Dynamic Bayesian classifier which factors the time history of features. Huang et al[40] used a probability-based model to demonstrate that prior knowledge of an environment can further improve Neural Machine Interface performance and lead to more stable transitions. However, "prior knowledge" of the environment was artificially achieved by altering probabilities by hand. Vision-based sensors were mentioned as possible ways to directly sense the environment and update these probabilities automatically.

Environment Sensing: Explicit vs Implicit

In a comprehensive review of assistive technologies, Tucker et al. [96] refer to methods involving Mechanical or EMG sensors as *implicit* environment sensing, because the controller builds an understanding of the environment by measuring user and device state. However, relying wholly on the user's state limits the possibilities of error detection and fault tolerance [32]. When the user reacts to an erroneous mode classification, their behavior can result in further classification errors. This can produce a propagation of errors and force the user to stop to reset the system. They are also limited in their ability to provide a long window into the future actions of the user and the upcoming environment changes. *Explicit* environment sensing could provide a parallel pathway to estimate the appropriate mode for the current terrain without relying only on user behavior. [40].

The ability to directly sense the environment, and anticipate transitions in advance would allow for triggering a change in controls at the right moment. Vision sensor is particularly suited for both of these requirements.

Direct Terrain Classification entails training a system to estimate terrain types (eg. stair, ramp) based on raw sensor data. This approach often requires manual calibration of the thresholding parameters, such as stair-rise or slope/incline of the ramp, needed to differentiate among terrain types. Huang et al. [58] have designed such a system to provide environment information via laser distance meter. However, the unidirectionality of the laser makes details such as the mounting height and the angle of inclination toward the floor critically important. The need for subject-specific calibration, such as stride length and device mounting height, also limits the wide-spread deployability of this system.

Krausz et al. [47] demonstrated scene segmentation using a depth camera for stair-ascent transition detection. Several practical difficulties with this approach were discussed by the authors including fixed height offset to detect floors, motion artifacts causing false detections during dynamic situations, and limitations caused by the angle of approach.

Advancements in vision sensors and miniaturization of powerful computing have allowed leveraging the improved performance of deep-learning for vision classification towards prosthetic control[48]. [62, 107] use depth sensors to classify terrains using support vector machines and finite state machine networks. [52, 117, 114] use deep neural networks on RGB camera data towards identifying terrain prior to kinematic changes are reflected on the user.

Vision for localization These studies demonstrate the potential benefits of obtaining explicit environment and terrain information directly from vision data. In contrast visual features can also be used towards estimating the location of a prosthesis in an environment. The location estimates cross-referenced with a map of terrain boundaries can be used for predicting current terrain and upcoming transitions. This localization based strategy was investigated as one of our primary aims and will be discussed more in detail in the following chapter.

2.2 Challenges of Mode-based Control

Having established a broad overview of mode-based control, we now discuss the challenges of this strategy. Given the vast space of possible movements, categorization of activities into a handful of "locomotion modes" comprising the most commonly encountered terrains and activities is an efficient technique. The gait cycle within a mode is further divided into phases (swing, stance), each with distinct controllers and control parameters. Control is transitioned between phases, within a mode, using a finite state machine approach. Each state uses a set of static parameters that are hard-coded into the controller. This approach quickly becomes unwieldy, as the number of tune-able variables rapidly increases with the number of parameters per control law, the number of states per activity, the number of activity modes, the number of joints to be actuated, and the number of limbs to be controlled. This has led to several independent studies exploring ways to manage [87] or using re-reinforcement learning to automatically tune these parameters [101].

2.2.1 Problem 1: Challenging transitions

EMG and mechanical sensors, most commonly used sensors for selecting the activity modes, are ideally suited to provide a direct window into the user state. They are able to understand the environment however is limited and indirectly derived through user actions (implicit environment sensing). Perceiving the environment indirectly compromises the fault tolerance as system tries to read the erroneous user state to rectify an error. In most cases, the user is forced to stop to reset the system. [32].

EMG activity can precede kinematic changes, but generally no more than 100ms [8]. Mechanical sensors offer even less future prediction horizons (Fig. 6.1). Late detection of terrain change precludes triggering mode transitions on time resulting in unsafe and choppy response. This has resulted in relatively worse performance during transitions. [31, 99]. With most walking bouts lasting less than 30 seconds [69], mode transitions are common and inconvenient [28]. Given these challenges, users often prefer passive devices for their simplicity while adopting compensatory movements which can be physically demanding and detrimental [49, 50].

Vision for environment sensing Humans use vision to look at upcoming terrains to anticipate changes and fluidly adapt to them. Even with wide variability of terrains, humans consistently looked 1.5 seconds ahead of their current location [63]. This is similar to look ahead timing seen in research on other motor actions—stair climbing suggesting that this timing plays an important role in human movement [72]. Computer vision is user-independent and allows future intent prediction upto 2 seconds in advance with high accuracy [114, 117].

To apply vision as an input sensor for prosthesis is not a trivial problem as very different visual scenes can have the same desired behaviour. A common approach is to train a classifier on images as inputs and corresponding mode labels as targets. Studies currently employ manual methods of labelling, which is time and resource consuming, even for modest number of modes [52, 62, 114, 117]. For example, [52] 37K vision frames hand labeled for single subject and 3 modes. There is subjectivity associated with manually labelling the environment classes, particularly near transitions[52, 62]. For transitioning from staircases to level-ground environments, the images were labelled as staircases whenever the staircase was visible inside the sampled field-of-view.

Data collection could be designed to aid labelling by restricting inputs from only one kind of terrain for any given session [114]. However, this omits the transitions which is the more challenging and needed scenario to be targeted.

As one of our primary aims we address the control challenges surrounding transitions using vision to provide explicit environment information. To label the vision without manual bias and effort, we introduce a method for choosing modes that arise from the data. We let the regularities present in natural movements dictate the number and characteristics of modes. These cluster labels constitute "modes," by virtue of their similarity in the data, instead of human assumptions. This allows for supervised training of a vision classifier, using cluster labels as classes. We discuss this more in detail in Chapter 6 pertaining to Aim 3b.

To summarize problem 1, explicit sensing of the environment with ability to predict future terrains will prevent the propagation of errors, improve fault tolerance, and transition performance. Vision is ideally suited for this, however, predicting terrains from vision requires large amount of hand labeled training data.

2.2.2 Problem 2: Unique and Non-Rhythmic Movements

The mode based control strategy is efficient to tackle the most commonly encountered activities involving rhythmic movements. However, physically active powered limb users cite challenges during unstructured activities that require unique non-rhythmic movements. Sports, getting in and out of cars or restaurant booths, obstacle avoidance, and navigating uneven terrain are examples of activities of daily living that do not cleanly correspond to commonly-used control modes.

A possible solution could be to define more modes to encompass more types of activities. This requires the construction of more mode-specific controllers and makes the task of choosing the 'right' mode more difficult. Moreover, construction of new modes requires labelled examples to train a machine learning model. This, as we discuss above, is time and resource consuming process prone to subjective bias.

A control strategy to address unstructured and non-rhythmic activities is necessary to fully leverage the benefits of powered limbs.

As part of our primary aims, we demonstrate two approaches to address the dearth of movements in current control strategies. We show a method to create new modes to augment the current mode-based strategy, which inherently relies on categorization of movements into finite classes. However, our method allows movements in the data to dictate the categorization and characteristics of the modes. This method also facilitates labeling of training data with the corresponding mode label. We discuss this methodology more in detail in Chapter 5 which elaborates Aim 3a.

Our second strategy is a data-driven controller that provides all desired movements, without categorization into modes. The controller applies a deep learning neural network to generate all the movements present in the training data. New movements can be added by re-training the underlying neural network with new activities. We discuss this approach in Chapter 4 which delves into Aim 2: Coordinated Movement Controller.

To summarize problem 2, unstructured and un-rhythmic movements desired by active powered limb users are currently not encompassed in the mode-based repertoire. Labelled training examples of movements are needed for increasing the number of modes. Alternatively, a data-driven continuous and unified controller can generate all the desired movements obviating modes and labelled examples.

Chapter 3

AIM 1: VISUAL LOCALIZATION FOR PROSTHETIC CONTROL

A major hurdle for the widespread use of wearable assistive devices is determining, moment-by-moment, the control mode appropriate for a given terrain when faced with a complex, multi-terrain environment. Current control strategies focus mainly on measurements of user behavior and less on environment information. This strategy has resulted in lower performance during terrain transitions.

The ability to anticipate transitions in advance would allow for triggering a change in controls at the right moment. EMG activity precedes a change in physical behavior with less than 100ms lead time [8]. For these reasons, vision as a sensing modality has recently garnered interest in research settings[112, 114, 107, 117, 52].

Our first approach addresses the problem of transitions by using vision for explicit environment sensing. We demonstrate the application of location estimates from a vision-based localization system to obtain environment awareness by delineating various terrains into regions. Given the current location and region occupied by the user, a controller could be built to select appropriate modes, predict transitions, or to add error correction.

We report the percentage accuracy of the classification and temporal accuracy in detecting terrain transitions. This approach improves fault tolerance by sensing the environment. The location of the prosthesis cross-referenced with known locations of transitions could be applied to anticipate upcoming transitions before the EMG or the body kinematics reflect the change.

3.1 Introduction

Assistive technologies have the potential to rehabilitate and drastically improve the quality of life of people with locomotive impairments. Prostheses and orthoses have existed in a mechanically passive form for several centuries. However, mechanically passive devices lack the capability of generating power, resulting in higher metabolic expenditure, uneven gait and increased stress to other joints [103].

Recent advances in technology (miniaturization of actuators and improved battery power, etc.) have enabled a new breed of lightweight assistive technologies far more capable than their predecessors. Next-generation devices use power or vary dynamic characteristics (eg. damping) to improve mobility, comfort, and stability [9], and can navigate a wide variety of terrains. They have terrain specific mechanical behavior profiles, known as locomotion modes, and several different control strategies have been employed to estimate the right

mode for any given moment.

Currently available methods for automatic mode selection are based on sensors on the device or user [99, 61]. They measure the mechanical state of the device or body, such as acceleration or forces sensed in the socket. In order to infer the intent of the user, there continues to be interest in using electromyogram (EMG) signals from the residual limb [41, 44]. A combination of EMG and mechanical sensors, Neuro-Mechanical Interface or NMI, has yielded an accuracy of 92% in real time [32]. A key finding was that additional sensing modalities can be combined to improve locomotion mode classification accuracy. This is important because while EMG shows great promise, it is also challenging to deploy due to the need for calibration, corrosion due to perspiration, crosstalk, electrode lift off, motion artifacts and interference with the socket [19, 57].

In a comprehensive review of assistive technologies, Tucker et al. [96] refer to methods involving Mechanical or EMG sensors as *implicit* environment sensing, because the controller builds an understanding of the environment by measuring user and device state. However, relying wholly on the user's state limits the possibilities of error detection and fault tolerance [32]. When the user reacts to an erroneous mode classification, their behavior can result in further classification errors. This can produce a propagation of errors and force the user to stop to reset the system.

Explicit environment sensing could provide a parallel pathway to estimate the appropriate mode for the current terrain without relying only on user behavior. Initial results from this strategy show improved mode classification accuracy when incorporating environment awareness [40].

3.1.1 Explicit Environment Sensing: Direct Terrain Classification

Direct Terrain Classification entails training a system to estimate terrain types (e.g, stair, ramp) based on raw sensor data. This approach often requires manual calibration of the thresholding parameters, such as stair-rise or slope/incline of ramp, needed to differentiate among terrain types. Huang et al. [58] have designed such a system to provide environment information via laser distance meter. However, the unidirectionality of the laser makes details such as the mounting height and the angle of inclination toward the floor critically important. The need for subject-specific calibration, such as stride length and device mounting height, also limits the wide-spread deployability of this system.

Krausz et al. [47] demonstrated scene segmentation using a depth camera for stair-ascent transition detection. Several practical difficulties with this approach were discussed by the authors including fixed height offset to detect floors, motion artifacts causing false detections during dynamic situations, and limitations caused by the angle of approach.

These studies demonstrate the potential benefits of obtaining explicit environment information via Direct Terrain classification. However, they also elucidate the wide gap that needs to be filled before direct sensing of of terrain is possible and can be safely applied to assistive technologies.

3.1.2 Explicit Environment Sensing: Localization

In this study we demonstrate explicit environment information via localization. Instead of directly mapping sensor data to terrain type, sensor data is used to estimate location, which can then be related to known regions of terrain. This method assumes that a map of the environment, cross-referenced with region boundaries, is available. For indoor environments, this has been a challenge in the past due to limited GPS availability and sensor inaccuracies. However, just as the last decade saw outdoor navigation transformed with Google Maps and GPS, recent research and technology trends suggest that such maps will be available in the near future [106, 5].

Several industry giants such as Google, Microsoft-Nokia, and Apple are building a platform for creating high-fidelity indoor maps[5, 1, 6]. Google Maps has uploaded floor plans of several multi-story buildings such as airports, malls, hospitals [2] and has also patented an indoor map server platform [75]. These indoor maps are equipped with annotations of common terrains such as ramps, or a staircase to the next level. Maps generated using LIDAR point cloud data are now within 0.05-0.07m accuracy[36] and several researchers have addressed automation of the process using robots.

Complementary to the mapping process, localization inside a mapped environment is also being tackled from several fronts with ever-increasing accuracy. In this context, localization is the task of determining the position and orientation of a device at any particular moment. Bluetooth, WiFi and magnetic field strength based navigation are active areas of research[65].

Simultaneous Localization And Mapping (SLAM) [90, 17] which combines localization and mapping under one umbrella, comes in a variety of flavors involving computer vision, Wifi signal strength, or a combination of all these[65]. In the last few years, robotics researchers have moved on from asking "Is SLAM possible?" to "Is SLAM solved?"[21]. Localization from SLAM is now effective, inexpensive, and likely to be built in to upcoming consumer electronics.

Since current localization systems are primarily designed for stable hand-held use, one of the goals of this study was to evaluate the performance of a localization system while mounted on the lower-limb, subject to rapid movements during walking. We focus on prosthetic control, but the localization strategy derived here translates to any assistive devices where environment information plays an important role.

We demonstrate two classifiers:

1. Instantaneous Region Classifier, which uses only the location estimates,

2. Time-history based Region Classifier - 1D Convolutional Neural Net, which uses a time history of location and height estimates.

We investigate the effect of increasing exposure to the environment and varying walking speed. We report region classification accuracy and the temporal error of the region transitions.

3.2 Methods

3.2.1 Localization System and Terrain

Multi-Terrain Course

The course consisted of flat ground, a platform reachable by stairs, and a ramp (Figure 3.1). Stairs were built to standard international residential specifications. Handrails were included on the raised platform as well as the stairs, at a standard 91 mm height. The course formed a rectangle approximately 8 meters by 5 meters, with the challenging terrain regions (stairs, platform, ramp) on the longer sides. Due to limited space availability there was no down ramp, and users stepped down one step to the floor.

Mounting

We developed a means for mounting the device on intact limbs using a standard sports shin guard and a custom 3D-printed mounting frame (Figure 3.1 inset). The mounting frame placed the device at approximately 0.4m +/-0.05m from the ground for all subjects. The frame was fabricated in ABS plastic using a rapid prototyping 3D Printer.

Localization System

We created a localization system which was comprised of Google's Project Tango Tablet[26], our custom Android application and region classifiers.

The Android application was built using the Project Tango's API to interact with on-board sensors, load pre-made maps (see Section 3.2.3), and generate location estimates. It then used the Robot Operating System (ROS) to publish and transmit the location estimates via wireless network to a base station for synchronization with the motion capture data. We fit the system with 7 retro-reflective markers and used a 12-camera infrared VICON motion capture system to measure its true location. We streamed the data from the motion capture system using ROS. Data was acquired from both systems at 60 Hz.

3.2.2 Classification

We investigated performance of two types of classifiers:

- Instantaneous Region Classifier
- Time-history based Region Classifier - 1D Convolutional Neural Network (CNN)

Bounded regions in the multi-terrain course were annotated by hand. The Instantaneous Region Classifier functions like a simple lookup table, combining the localization system's position estimates with the region annotations to generate instantaneous region labels (Figure 3.2).

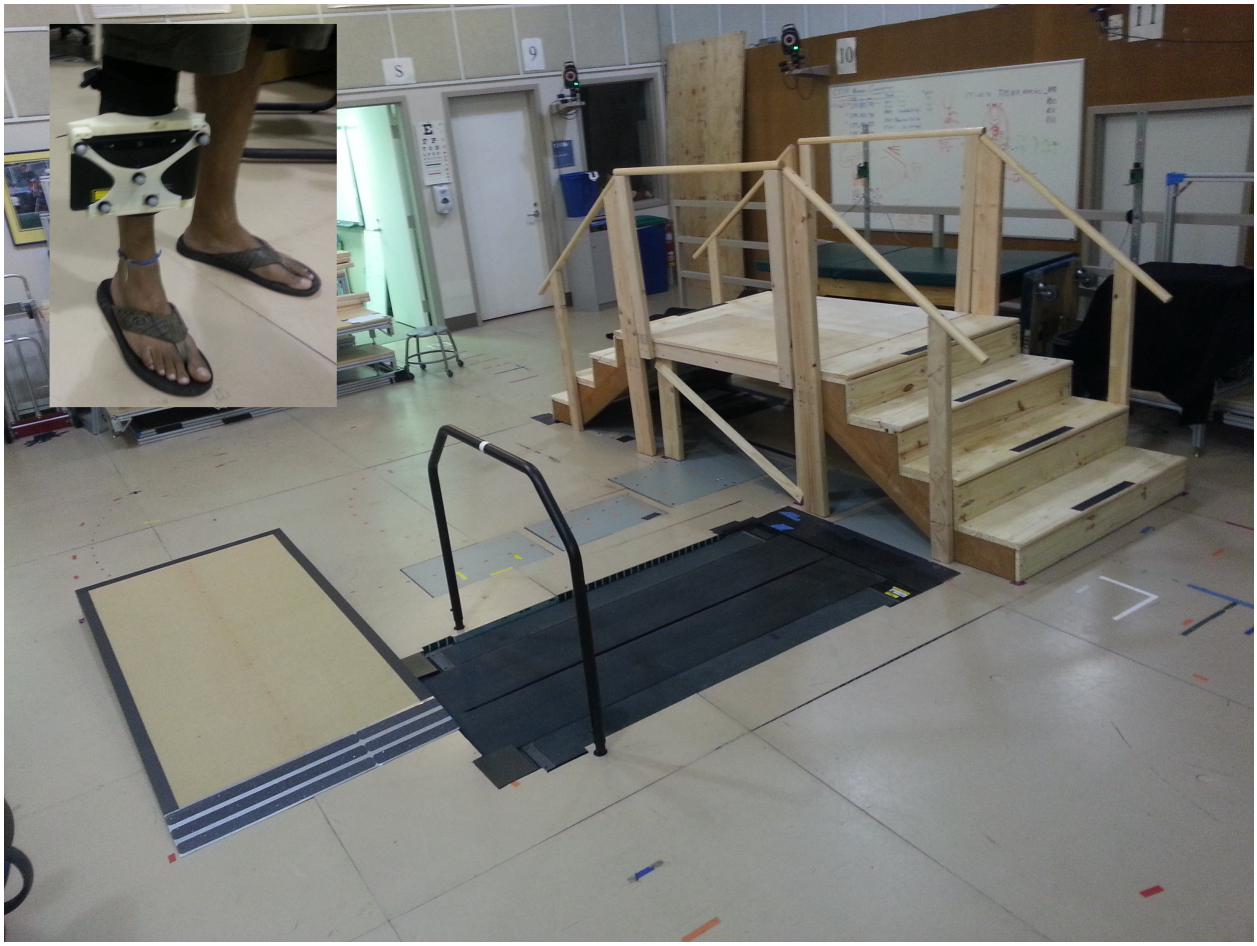


Figure 3.1: **Multi terrain course.** The 3D-printed mounting frame (inset) integrates with a standard shinguard worn on the shank of the right leg. Participants walked in a rectanglar course including up stairs, flat platform, down stairs, flat ground, and a gentle ramp. The black treadmill in the center was not used in the experiment.

The feature vector for instantaneous region classifier was

$$f_{region} = (X, Y)$$

The time history based region classifier incorporated the time history of location and height estimates, using a 1D Convolutional Neural Network (CNN) architecture. Selection of CNN was based on its success in capturing time-varying characteristics of input time series without the need for hand-crafted featurization[108]. The network was implemented on TensorFlow. Hyperparameters for the network, such as the number of filters and depth of the network, were obtained by trial and error and the best performing parameters were

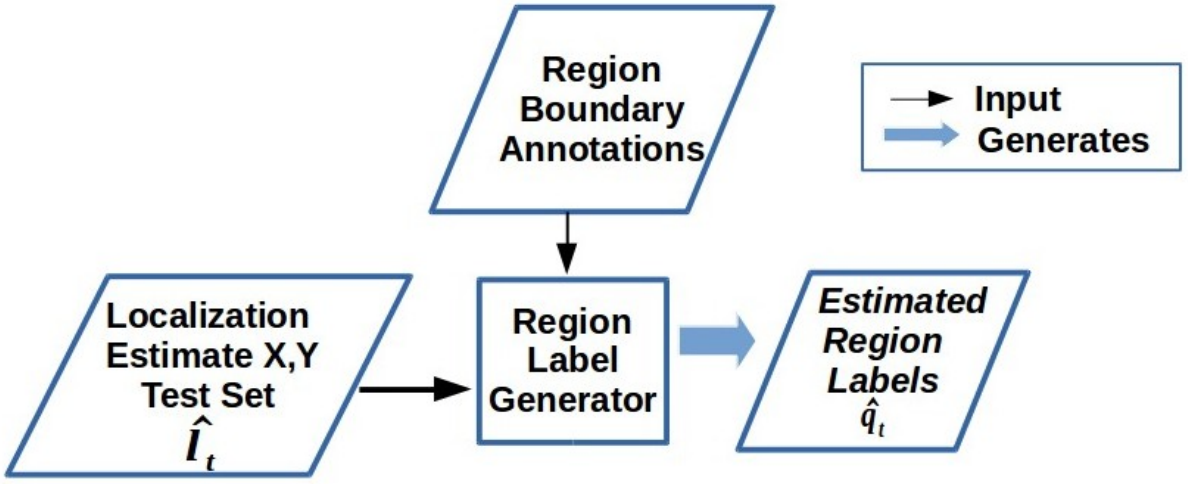


Figure 3.2: **Instantaneous region classification procedure.** Parallelograms denote data, and squares denote operations. The instantaneous region classifier only uses X,Y position data and regional boundary annotations for estimating instantaneous region label.

used to report final performance. The time history was limited to 20 time steps (320ms), which is less than half a gait cycle, to avoid the possibility that the particular sequence of terrains in our course could be leveraged for classification.

For testing, data from one trial is used as a test dataset and data from all other trials of the same subject and mapping condition are used as a training dataset. That is, if a trial from *Comfortable* Speed with *MAP4* is used as test data, *Comfortable* **and** *Fast* paced trials with *MAP4* are used for training. The process is repeated such that all trials are used as test sets and an average accuracy is reported as the Region Classification Accuracy (see 3.3.2) for that subject.

Figure 3.3 shows the general procedure for training the CNN classifier. Using the ground truth of user location from Vicon X and Y data, along with prior region annotations, we generate true region labels q_t for each time step t for all trials in the training set. We then use LOOCV to train a classifier with localization system's X,Y and Z estimates as features, with the corresponding true region labels, to generate a classifier model. This model can take a new test feature set of X, Y and Z data, and for each time step predicts a region label $\hat{q}_t$. The trial that was omitted in the training set is used as test input and region prediction is generated. The feature vector for time-history based classifier was

$$f_{learning} = (X, Y, Z)_t$$

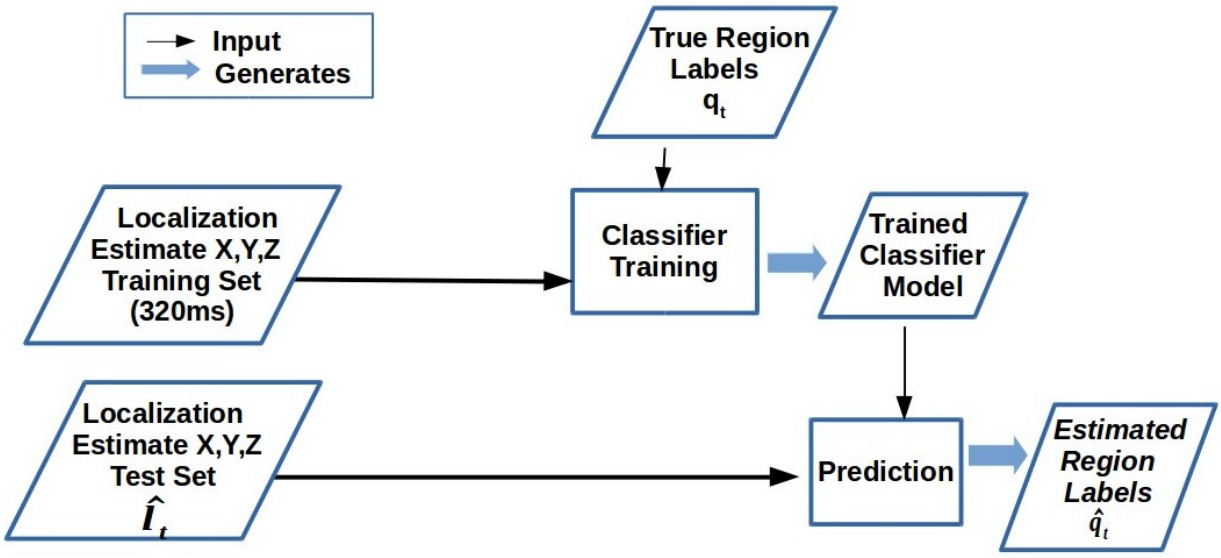


Figure 3.3: **Time-history based training and classification procedure.** Time-history based CNN classifier used a short time history of X,Y position data as well as Z direction height variations to learn and predict region labels

3.2.3 Experiments

Experiments were designed to evaluate performance with respect to change in exposure to environment and walking speeds.

Speed Conditions

The speed conditions were based on the instructions spoken at the beginning of the block of 10 trials. The participant was either requested to walk "at a *Fast* pace" through the course, or "at a *Comfortable* pace." These two conditions resulted in average walking speeds of 1.67 m/s and 1.15 m/s.

Mapping conditions

To provide an estimate of absolute location in a particular known area, also known as localization, the system requires a pre-made map. Creating a new map requires walking around the environment, exposing the camera to views as you walk. These maps are then processed and saved by the localization system using Tango API and the underlying Placeless-Place algorithm[60], to extract movement features from views. Our system then allows us to load the maps whenever revisiting the pre-mapped environment, to generate location estimates.

It is also able to send data (location estimates) as ROS messages over WiFi, synchronize with motion capture, and output region labels.

In order to assess the effect of additional exposure to the environment, we created three separate maps using 1, 4, and 8 laps around the course at comfortable pace. These maps are referred to as *MAP1*, *MAP4*, and *MAP8*. Hence, *MAP8* has eight times the exposure to the environment compared to *MAP1*.

The origin and the coordinate frames of the localization system and VICON motion capture system were matched for comparison without having to estimate the transformation.

Ambulation data was collected for 8 healthy participants (one female, median age of 25) with no amputation or other mobility impairments. The participant wore the localization system on the shank of the right leg and walked clockwise around the course (Figure 3.1). Trials were repeated for two speed conditions and three mapping conditions. Each subject performed 10 trials of these 6 possible combinations of speed and map, resulting in a total of 60 trials per subject. The starting position and order of trials was randomly varied to prevent bias. The experiment was completed in a single session which lasted less than 2 hours. Recruitment and human subject protocols were performed in accordance with local VA Institutional Review Board approval and each subject provided informed consent. De-identified data can be made available, via a data use agreement, upon request to the authors.

3.3 Analysis and Performance Measures

To assess the efficacy of the system in a subset of real-world scenarios, we quantified its ability to estimate the location of the user, and how well these location estimates could be used to estimate current region. We report 1) Localization Error, 2) Region Classification Accuracy, and 3) Absolute Temporal Error of region transitions.

3.3.1 Localization Error

Localization Error is simply the error in location estimates, as compared to measured ground truth, in meters. It is a measure of overall positional accuracy of the system.

Consider the location estimate from the mobile system to be $\hat{l_t}$ and the ground truth location from the motion capture to be l_t, for time t = 1 ... T. We define the Localization Error to be standard root mean squared error (RMSE) for a single trial:

$$\text{Localization Error} = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(\hat{l_t} - l_t)^2}$$

3.3.2 Region Classification Accuracy (RCA)

Region classification is a key output of the system: it is the percentage of accurate classification of the moment-to moment region labels based on location estimates. This requires

that a known mapping of region boundaries. Drawbacks and challenges of this requirement are addressed in the discussion section. The estimated region label is generated from the methods described in Section 3.2.2 and the correct region label is known from the motion capture ground truth.

$$\text{RCA} = \frac{\text{Number of accurately classified time steps}}{\text{Total Number classified timesteps}} \times 100\%,$$

This capability alone could be used to switch control modes of an assistive device, or used in conjunction with other control switching methods to improve accuracy and fault tolerance. For example, Figure 3.4, the orange box indicates a *Platform* region. When the system estimates that the user is in that region, a flat ground walking mode could be used.

3.3.3 Region Transition Detection

There are 6 region transitions encountered in this study. Three are transitions from Level ground walking (W) to Ramp Up (R), Stair Ascent (SA) and Stair Descent (SD) and three are exactly opposite transitions ($R \rightarrow W, SA \rightarrow W, SD \rightarrow W$). Ramp Down section was omitted to maintain clear separation of terrains with a distinct Level ground walking section. Inadequate positional accuracy could sometimes estimate the location of the system to be outside the regional boundaries resulting in missed or delayed transitions.

The Temporal Error of a transition is defined as the time difference between the ground truth location entering a new region and the estimate entering that new region. The average Temporal Error for a trial is the average of all transitions' Temporal Error for that trial.

It should be noted here that Temporal Error provides a measure of the localization system's temporal mismatch in detecting transitions, with respect to ground truth (motion capture) and not its ability to predict such transitions in advance.

Temporal Error is computed in the following way:

Each location estimate from the mobile localization system, denoted $\hat{l}_t$ for timestep $t = 1...T$ is assigned a region label $\hat{q}_t$, obtained by methods described in Section 3.2.2. In this case take values on 0, 1, 2, 3, 4, corresponding to the five different regions.

A change in region label between successive timesteps (e.g, $\hat{q}_{t+1} \neq \hat{q}_t$) is considered a region transition. For the location estimates, these are detected transitions.

Each timestep t has an associated time from the system clock, and we record the time for each occurrence of a transition as defined above. Denote these times $\hat{c}_t, c_t$ for estimated and true transition, respectively. Then the Temporal Error (T.E.) of that transition is

$$T.E. = |\hat{c}_t - c_t|$$

.

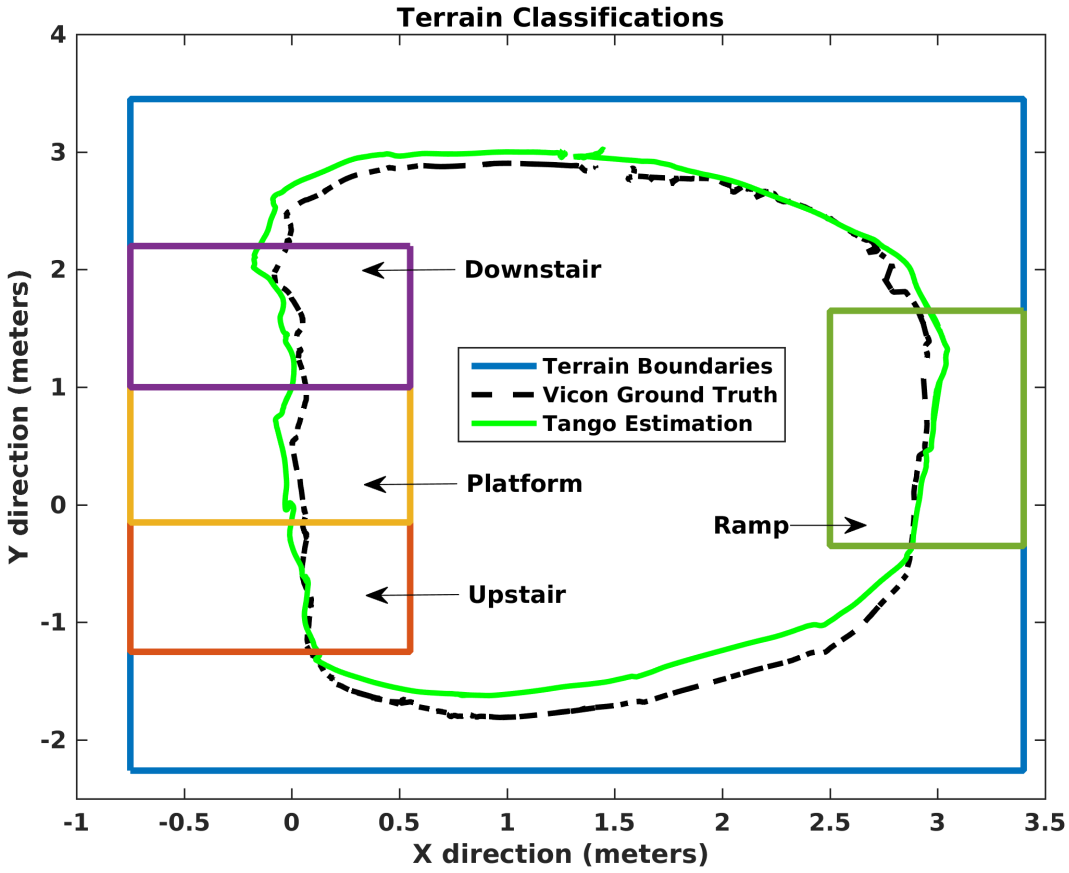


Figure 3.4: **Example location data for a single trial from the VICON motion capture system and the mobile localization device.** This trajectory, which is representative of most trials, is from a trial in the "*Fast*" walking condition using a MAP8, i.e. it was based on 8 laps. This trial had a mean localization error of 0.09m, and the average for this speed and mapping condition was 0.11m.

3.4 Results

The system was able to successfully estimate location, provide terrain awareness, and detect region transitions as the user traversed the course.

3.4.1 Localization Performance

The mean squared localization error decreased inversely with more exposure to the environment; this indicated higher fidelity to the ground truth location. (Figure 3.5) The im-

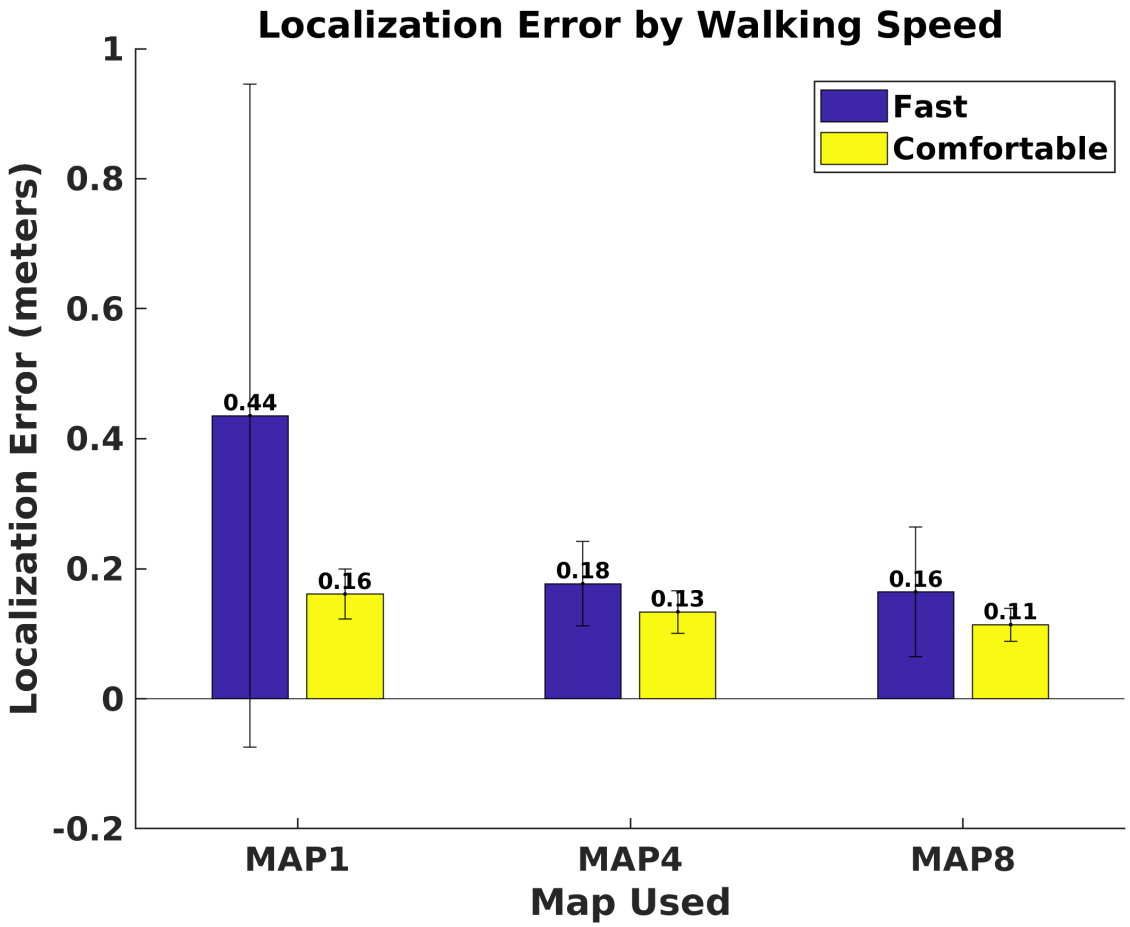


Figure 3.5: Mean localization error and standard deviations for trials. Localization error generally decreased for slower walking (blue vs yellow bars) and more laps used to create the map (bar groups).

provement in localization error between *MAP1* and *MAP4* was more pronounced than when increasing to *MAP8*. Higher exposure maps were also more resilient to errors caused by faster walking pace.

3.4.2 Region Classification Accuracy

Location estimate accuracy and hence Region Classification Accuracy was improved when using maps with more exposure than 1 lap (*MAP1*) but did not show significant improvement beyond an exposure of 4 laps (*MAP4*). The time history classifier exhibited improved accuracy in all cases (Figure 3.6 and Figure 3.7).

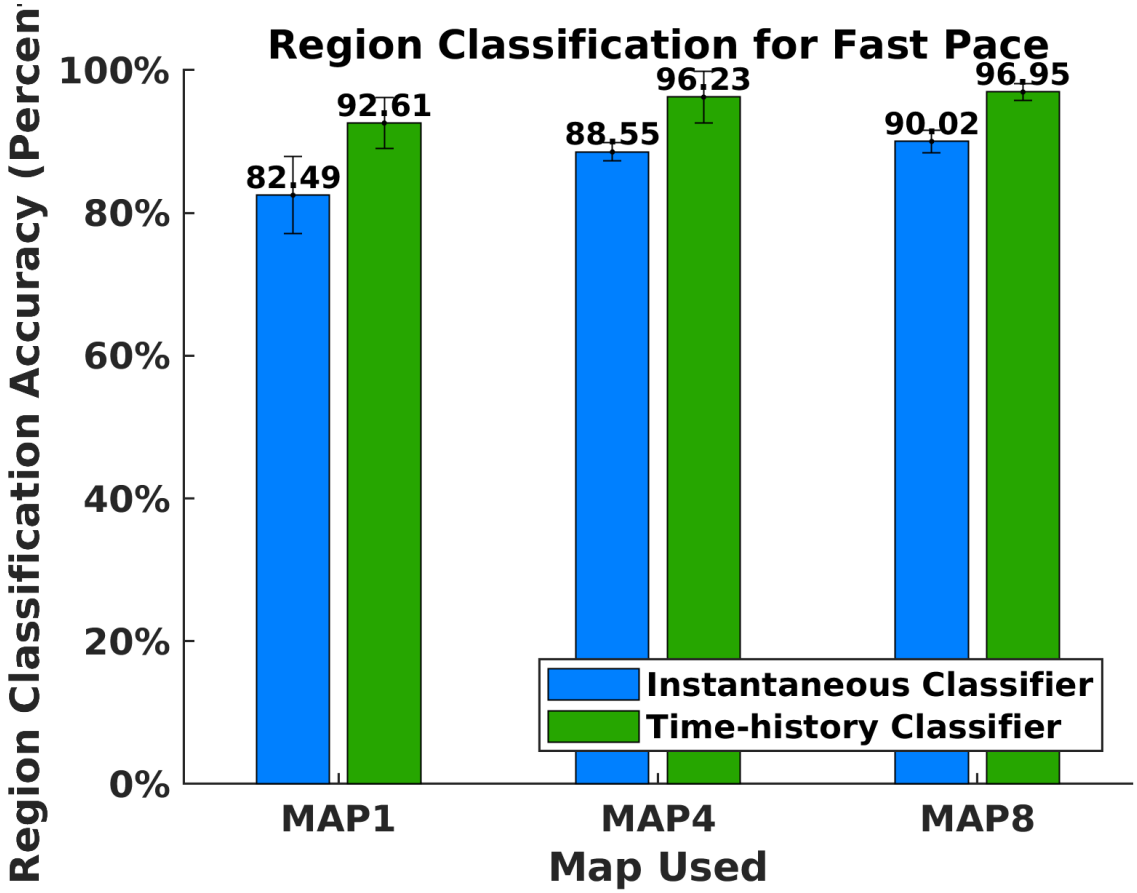


Figure 3.6: **Region Classification Accuracy for *Fast* speed condition.** Region Classification Accuracy (RCA) is the measure of the percentage of time the system correctly estimated the current region. Time-history of locations improved accuracy for all maps.

3.4.3 Transition Detection Performance

All transitions were accurately detected using *MAP4* and *MAP8* maps and *Comfortable* speed for both classifiers. The time history based classifier exhibited faster detection of transitions for all cases. Average Temporal Error decreased with maps having more than 1 lap of exposure, however increased exposure beyond 4 laps did not seem to result in better performance for detecting terrain transitions (Figure 3.8 and Figure 3.9).

3.5 Discussion

Our study demonstrates that there exist mature, ready-to-use localization technologies that can be adapted for environment awareness.

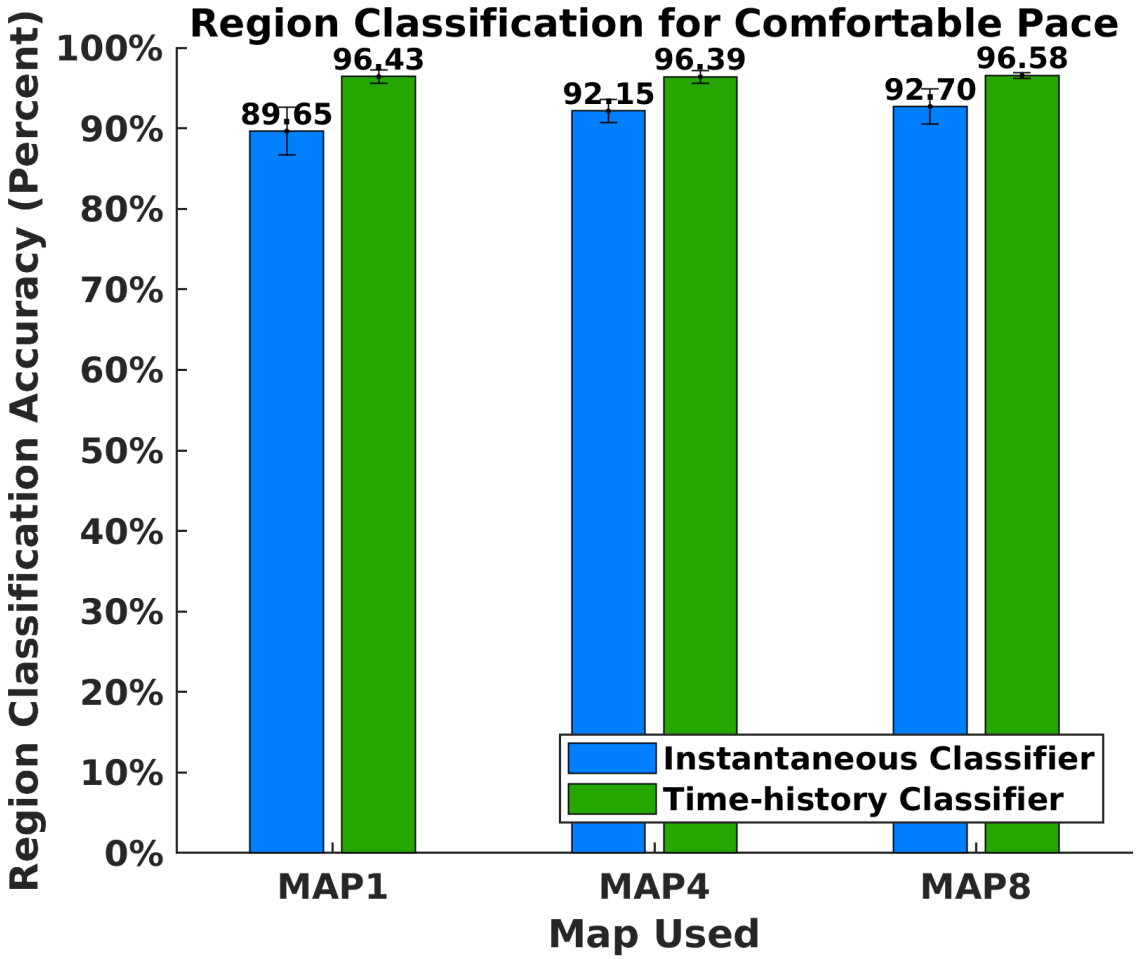


Figure 3.7: **Region Classification Accuracy for *Comfortable* speed condition.** Region Classification Accuracy improved with more exposure for *Comfortable* speed as well. For each type of classifier, performance was about the same for both speeds.

Performance improved with increased exposure to the environment, but 4 exposures was sufficient to obtain near peak performance. Learning the time-ordered sequence of location estimates resulted in near perfect (96%) region classification accuracy and detected transitions with a maximum delay of 110ms. This could be because there are idiosyncrasies or unique temporal patterns present in the gait, that provide more clues about the region than simply the location estimate. Performance was also robust to variations in speed.

These results show that locomotion mode estimation using only localization can have comparable results to using only mechanical sensors [99] or only EMG sensors [41]. However, we expect localization integrated with other sensor modalities to improve performance, as

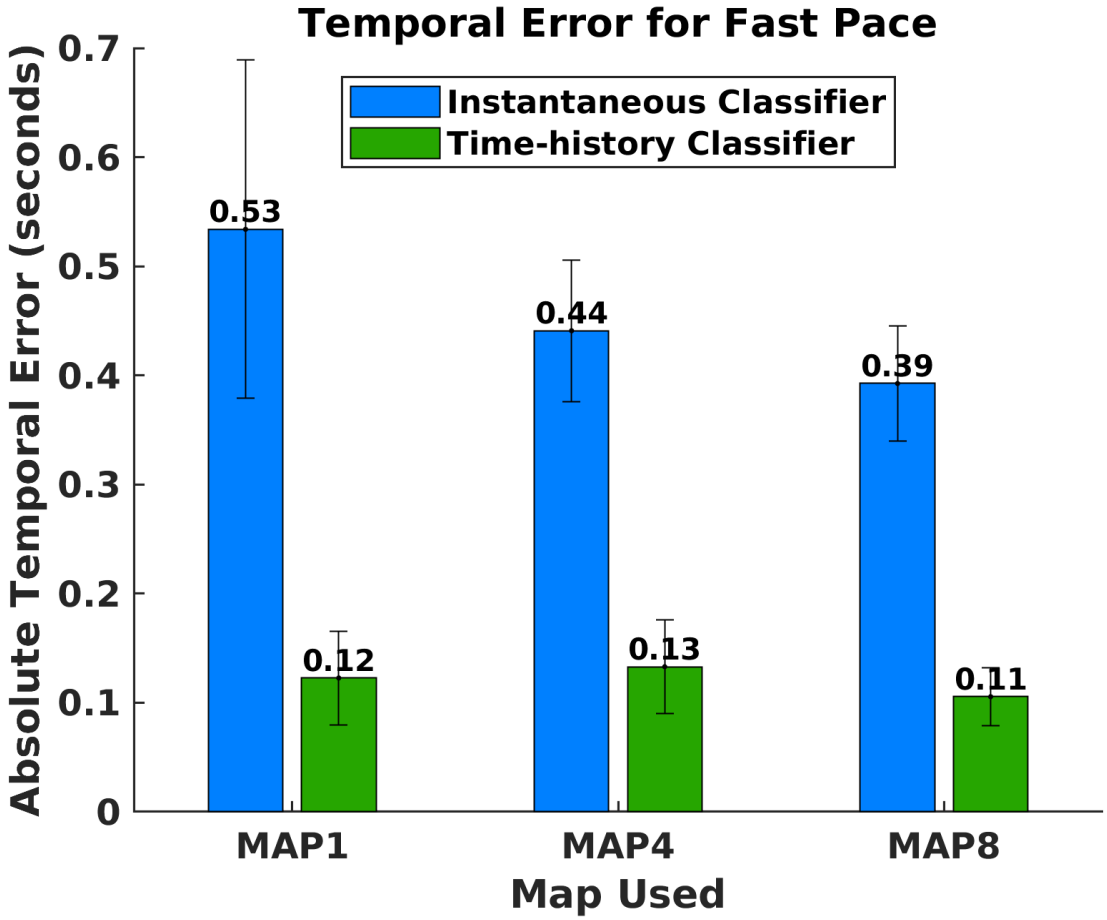


Figure 3.8: **The Temporal Error of transition detections is the difference in time between true and estimated terrain transitions.** All the transitions detected by the time history classifier were within 110 milliseconds for the *Fast* speed condition. For both classifiers, exposure beyond 4 laps in the map did not produce significant improvement. Error bars denote standard deviation for all trials under that condition.

seen in Neuro-Mechanical Interface (NMI). A system incorporating multiple methods could use redundancy to make a more robust controller.

Environment awareness via localization can enable error-correction and even anticipatory control. The temporal error of 110ms was the maximum system delay when the classifier model was trained to estimate *current* region. However, for anticipatory control the classifier could be trained to predict the up-coming region. The controller would use localization data to estimate the user's relative distance from an upcoming region transition and seamlessly switch modes, perhaps even before the user's behavior begins to signal the transition.

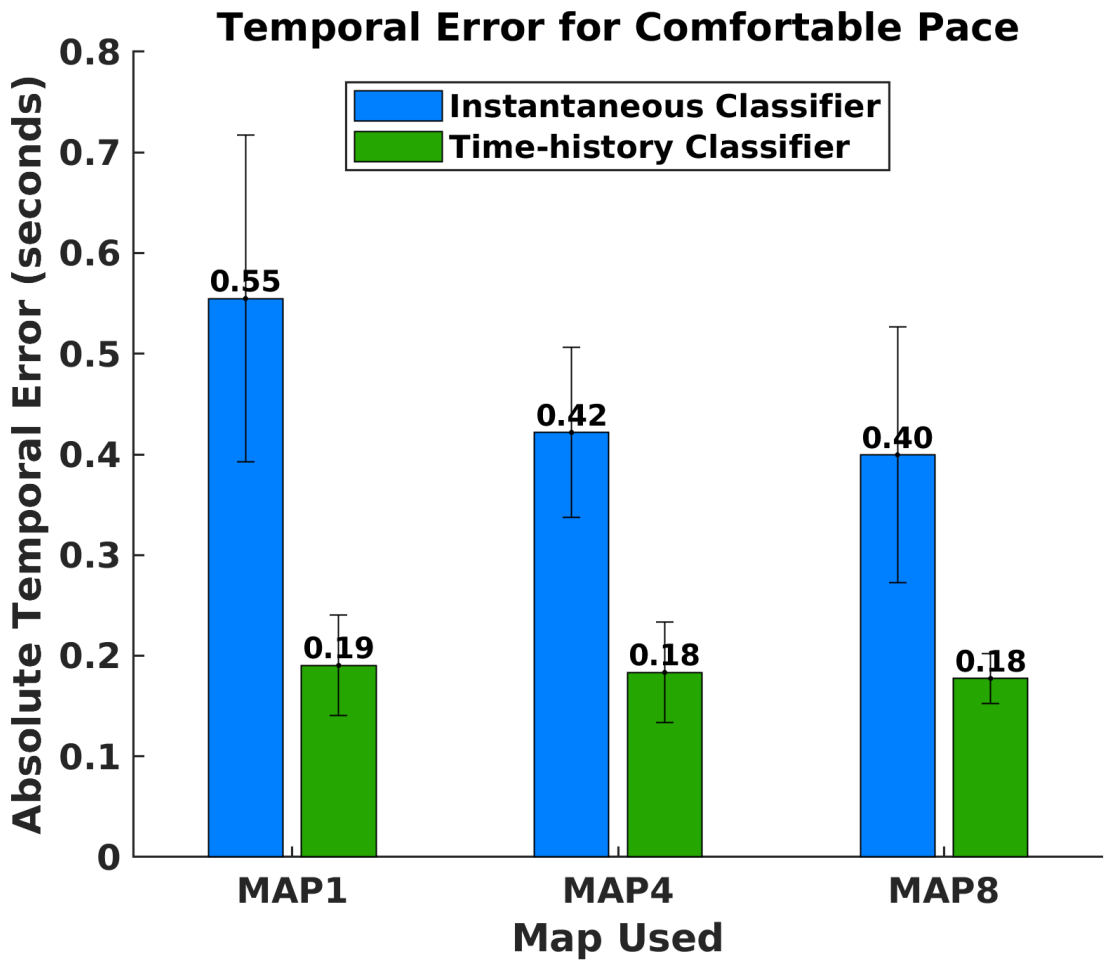


Figure 3.9: **Temporal Error for *Comfortable* speed condition.** Similarly to the faster walking pace, Temporal Error improved with exposure from 1 to 4 laps, and was much lower for the time history classifier.

The Instantaneous Classifier used only region annotations and required no subject-specific calibration or training. The experiments were performed for different subjects using the same maps on different days. Perhaps this is one of the most promising implications of this study. Robustness to subject-specific factors like walking style, height and even mounting location could be leveraged for wide-spread applicability.

The disadvantage of localization is that it requires a map of regions to cross-reference against the estimated location. It is a significant feat to obtain annotated maps of environments, but it is one that has been surmounted before for outdoor environments.

As a representative platform we used the Project Tango device [26], a SLAM based

technology, which uses computer vision, inertial sensors and the Placeless-place recognition algorithm[60] for localization. However, the benefits of using indoor localization for environment awareness are not contingent on the particular sensor, but may be realized using whichever of the many upcoming systems (WifiSLAM, Bluetooth etc.) emerge as standards. Similarly, maps may be sourced by any of the methods discussed in Section 3.1.

The system we describe here has been designed to seamlessly plug into research robotic assistive prototypes. Experiments in preparation will repeat the protocol for participants wearing assistive devices. We expect gimbal based stabilization could reduce motion artifacts and improve performance. These further experiments will allow detailed examination of what causes errors and how to optimize the system.

3.6 Conclusion

A key motivator of this research is that localization is appearing in a variety of consumer applications, and we expect its market penetration to be high, similar to how Global Positioning Systems have reached ubiquity[5]. While early results made use of specialized equipment and restrictive assumptions, the barriers to wide inexpensive deployment have steadily been broken. The study we describe here shows that visual and inertial sensors mounted only on the leg can be used for localization, and that this localization is plausible for locomotion mode estimation. Leveraging this rapidly improving technology for rehabilitation could unlock the full benefits of powered assistive technologies.

Chapter 4

AIM 2: COORDINATED MOVEMENT FOR CONTINUOUS CONTROL OF PROSTHETIC LIMBS

Our second aim illustrates the data-driven CM controller which generates a *continuous* prosthetic reference trajectory, eliminating the need for the discretization of activities into modes and explicit transitions from one mode to another (Fig 4.5). The CM controller is *continuous* because it uses no concept of discrete gait events or phase, and *unified* because it is a single controller that provides all desired movements, without switching modes. Previous attempts [84, 78] at continuous prosthetic control have aimed at unifying the gait phases (swing, stance, etc) but within the same mode (flatground). The CM control strategy achieves continuity across gait phases and different gait modes.

We exploit the strong inter-joint coordination of human movement to predict the kinematic trajectory of a single joint based on the movement of the rest of the body. This predicted trajectory could be used as a reference to control a prosthetic actuator joint in real-time.

We report performance for structured rhythmic activities such as flatground walking, and unstructured activities demanding unique non-rhythmic activities (side shuffles, weaving through cones, backward walking).

4.1 Introduction

Powered limbs are getting more and more capable in their hardware capacity[10, 55]. The new devices are lighter, more powerful with multiple joints and degrees of freedom. Given the vast space of possible movements, intuitive control of powered limbs however is still challenging. **Categorization of activities into a handful of "locomotion modes" comprising the most commonly encountered terrains and activities such as flatground walking, stair ascent, etc. has been an effective and prominent strategy** [32, 41, 99]. The selection of mode for any given instant is done using sensor data and machine learning models (SVM, LDA, or ANN, etc). The current state of the art uses a combination of EMG and mechanical sensors as inputs to understand user intent *and* environment needs.

Powered lower limb users cite inability to use during unstructured activities that require unique non-repeated movements as practical limitations[28]. Sports, getting in and out of cars or restaurant booths, obstacle avoidance, and navigating uneven terrain are examples of activities of daily living that can be challenging and do not cleanly correspond to commonly-used control modes. Moreover, with most walking bouts lasting less than 30 seconds [69], mode transitions are common. Users often prefer passive devices for their simplicity while

adopting compensatory movements which can be physically demanding and detrimental [49, 50].

A control strategy to address unstructured and non-rhythmic activities is necessary to fully leverage the benefits of powered limbs. A possible solution could be to define more modes to encompass more types of activities. This requires the construction of more mode-specific controllers, and makes the task of choosing the 'right' mode more difficult.

Another potential limitation on expanding the number of modes is the tuning of the control parameters which go with it. In current mode-based control, the gait cycle within a mode is divided into phases (swing, stance), each with distinct controllers and control parameters. Control is transitioned between phases, within a mode, using a finite state machine approach. Each state uses a set of static parameters that are hard-coded into the controller. This approach quickly becomes unwieldy, as the number of tune-able variables rapidly increases with the number of parameters per control law, the number of states per activity, the number of activity modes, the number of joints to be actuated, and the number of limbs to be controlled. This has led to several independent studies exploring ways to manage [87] or using reinforcement learning to automatically tune these parameters [101].

Another weakness of the phase-based strategy is that it cannot handle abrupt mid-phase changes in activities. Each gait cycle must begin with the particular gait-phase (for e.g heel-strike). This strategy also inflexible to accommodate irregular or unprogrammed gait patterns (e.g. walking in a crowd or over rocky ground), unexpected events such as tripping. A non-phase based control has been the topic of some studies to overcome the limitations of phase-based control.

Exploiting Whole-body Coordination for Non-phase based Control

A few studies have explored using a non-phase based continuous control approach to reduce the number of tuned parameters and address non-rhythmic movements. Instead of detecting gait phases within a gait cycle, most non-phased control studies rely on strong inter-joint coordination exhibited by locomotion[12] to provide a continuous movement. This coordination of the whole body means that movements of any one joint are highly correlated with the movement of the rest of the body.The trajectory of movements of the intact limbs provide the means to estimate the movement of the missing limb with a high likeness to observed behavior. This observation motivates our Coordinated Movement controller which will be described in the following subsection.

An earlier approach called echo control replayed or "echoed" the movement of the sound leg on the prosthetic limb[27]. However, this delayed playback failed when asymmetric movements were desired and required the sound leg to lead all movements.

Complementary Limb Motion Estimation (CLME) infers the intended motion of affected limbs from the motion of the residual limbs, and maps this to a reference trajectory for robotic prosthetic joints to track[97]. The mapping is derived through regression of physiological gait recordings of healthy subjects.

A more recent success in non-phase based control strategy uses virtual constraints to define joint trajectories as functions of a monotonic phase variable that continuously represents the gait cycle across the entire stride[78]. The phase variable is computed from residual thigh motion, giving the amputee control over the timing of the prosthetic joint patterns. This strategy reduces the number of tuneable parameters. The holonomic nature of human gait, and the user-controlled progression of control allows performing activities such as backward walking and walking over obstacles. [84].

The strategies are 'continuous' in that they do not divide the gait cycle into sub-phases. They do not, however, provide continuous control over *different modes* and transitions. They do reduce the tuneable sub-phase control parameters but do not unify movements of different modes or activities. Each mode requires tuning of mode-specific parameters [84] separately. CLME was demonstrated on rhythmic activities of flatground walking and stair ambulation, but left non-rhythmic activities as future work. [84, 78] address more non-rhythmic activities, such as obstacle crossing, but yet the activities were holonomic functions of flatground walking. More unstructured activities like side-stepping and weaving around obstacles are yet to be addressed for prosthetic control.

Our Coordinated Movement (CM) Controller

Towards providing continuous control across gait phases and activities, we describe the coordinated movement controller applying whole body kinematics and powerful deep neural networks. Our data-driven approach is trained on locomotion data collected from various activities to predict kinematics of a target joint omitted from the input set of joints. For example, in the case of a trans-tibial prosthetic application this target joint would be the ankle joint. The body motion of the person with the amputation would serve as the inputs and the predictions generated by the network could be used for the control of the prosthesis.

Recent advances in machine learning which leverage the availability of large datasets rather than employing carefully human-engineered rules. “Learning by examples” has surpassed all previous “learning by rules” efforts in a variety of applications, such as speech recognition and stock-market prediction [29].

Several studies have explored deep learning paradigms to solve the problem of human activity recognition (HAR) without the need for hand-crafted featurization. [94] [43]. Most of the studies have focused on the categorical classification of human activities such as sitting, running etc and are trained on well-defined chunks of time-series. This strategy can have difficulty ”scaling up” to recognize complex high-level behavior which can have hour-long to day-long duration [68]. *Non-categorical* control with *continuous* joint trajectory generation for prosthetic limb control has yet to be fully explored.

In [79], we demonstrated a proof of concept, generating ankle joint trajectories, but for structured activities (flat ground walking and stair ambulation.) In [80] we expanded the envelope of operation to include non-rhythmic movements and agile maneuvers. In this dissertation, we describe an expanded system to include the knee and ankle simultaneously. We describe here a real-time controller to acquire live kinematics and actuate a powered

knee and ankle prosthesis [10] on a benchtop platform. In the chapter 7 we demonstrate the controller with a subject wearing the prosthesis

4.2 Methods

4.2.1 Participants

Ambulation data was collected for a total of 65 healthy participants (34 male, median age 25) with no amputation or other mobility impairments. Recruitment and human subject protocols were performed in accordance with the local University of Washington Institutional Review Board approval and each subject provided informed consent. De-identified data can be made available, via a data use agreement, upon request to the authors.

11 subjects performed flat ground walking activity and 44 subjects performed stairs activity. To investigate atypical and non-rhythmic movements, 10 subjects performed 3 activities from the Comprehensive High-Level Activity Mobility Predictor (CHAMP) test. The CHAMP test was designed as a safe performance-based measure of high-level mobility for those with lower limb loss (See Section 4.2.2).

4.2.2 Activities

The movements targeted in this study were designed to be more challenging and difficult to be categorized into modes.

Flat ground

To replicate regular community ambulation, flat ground activity consisted of walking on a long corridor in a public building. This included random stopping to incorporate transitions between steady state walking and rest.

Stairs

This activity consisted of stair ascent and descent in a 6-story public building. This included sections of flat ground transitions in between levels.

Comprehensive High-Level Activity Mobility Predictor (CHAMP)

The CHAMP test was designed as a safe performance-based measure of high-level mobility for those with lower limb loss (LLL). The CHAMP test consists of 5 activity sets, but here we selected a subset of 3 activities that focus on agile movement, as opposed to balance or endurance. These are the Edgren Side Step, the Illinois Agility Test, and the T-test. They include challenging abrupt changes in movement direction, running, and backward locomotion. These standard tests are described in detail in [23, 81] and summarized below.

Edgren Side Step : Five cones are placed in a line three feet apart. The participant, starting from the center cone, sidesteps to the right until their right foot crosses the outside cone. The participant then sidesteps to the left until their left foot crosses the left outside cone. The participant sidesteps back and forth to the outside cones for 10 seconds.

Illinois Agility Test : The aim of the test is to complete a weaving running course in the shortest possible time. The length of the course is 10 meters and the width is 5 meters and cones mark the course. On the 'Go' command, the subject runs the course, without knocking down any cones.

T-test : A course 10 meters long and 3.5 meters wide is marked by cones in a 'T' shape. Successful navigation requires side shuffling to reach the left and right most cones as well as backward walking to return to the starting location.

4.2.3 Experiments

The subject's anthropometric details were recorded and 17 wearable sensors were placed on their body on locations as shown in Fig.4.1. This was followed by a calibration procedure and brief test to see quality of data being recorded. The subjects then performed 10-15 minute trials of the desired activity. For flat ground and stairs activity, the subjects were instructed to walk naturally at self-selected pace. The data for these activities was collected in public spaces during active business hours, with the intent that normal gait dynamics and corrections would appear in the example data. The order of activities, starting and ending points were randomized. The experiment was completed in a single session which lasted less than 2 hours. In total, 750 minutes of data were collected from all subjects.

4.2.4 Instrumentation

We collected locomotion data using the Xsens Awinda suit [4], consisting of 17 body-worn sensors placed at key locations. Each sensor has a tri-axial gyroscope, accelerometer, magnetometer, and barometer. Xsens Analyse software integrates these individual sensors and renders a full-body avatar. After a system specified calibration, the software provides position and joint kinematics in a 3D environment. Although other data such as limb-segment position, orientation, acceleration are available, we used only joint angles for this study. All angles are in 1x3 Euler representation of the joint angle vector (x, y, z) in degrees, calculated using the Euler sequence ZXY using the International Society of Biomechanics standard joint angle coordinate system [105]. Data, sampled at 60 Hz, from a total of 22 joints in 3 anatomical planes (sagittal, frontal, transverse) were captured for each trial, which results in 66 total possible features for our machine learning methods.

4.2.5 Data Processing

Sensor data was visually inspected to detect any potential equipment malfunctions or calibration problems. Sensors getting displaced from their original calibrated location is a commonly seen issue with wearable motion capture systems. If this was detected during the experiment, sensor placement was corrected followed by recalibration and reinitialization of the suit.

Xsens features a real-time engine that processes raw sensor data for each frame, algorithmically fits the human body model to estimate anthropomorphic joint and segment data. A post processing engine includes information from the past, present, and future to get an optimal estimate of the position and orientation of each segment. This 'HD' processing raises the data quality by extracting more information from larger time windows and modeling for skin artifacts etc, but also takes significantly longer time.

We used HD processed data as training data for our neural networks. The real-time benchtop test, however, used only real-time (non-HD processed) data.

4.2.6 Machine Learning Model and Architecture

Previously [79], we investigated three different regression models: Multivariate Linear Regression, Fully-Connected Deep Neural Network, and a variant of Recurrent Neural Network - Long Short-Term Memory (LSTM) [35]. It was shown that the Recurrent Neural Network with a short time-history of gait movements provided the best prediction of the right ankle joint. In this study, we use the same network architecture, shown in Fig.4.1. However, we increase the number of joints predicted to two by including knee joint as well. The deep-learning networks were implemented on PyTorch [71].

In the real-time prosthetic controller, the trained network would be applied to predict joint trajectories for a user whose movements would not have been captured in the training dataset. To analyze performance, one of the subjects was omitted from the training set. A random trial from this subject was used as the validation set, and another was used as the test set for reporting results.

Given a time series trajectory of M intact joints $x \in \Re^{M \times T-1}$, we employ the LSTM network model to estimate current target joint values $\hat{y_T}$ at time instant T.

$$\hat{y_T} = f(x) \tag{4.1}$$

where f is the LSTM network.

Data Normalization and Reshaping Each of the joint angles exhibits a different Range of Motion (ROM). In order to prevent high-ROM joints from dominating predictions, it is common practice to normalize all features (generally 0 to 1). We normalized all joint angles for every trial and saved the average scaling factor of the training samples for de-normalizing the predicted joint angles.

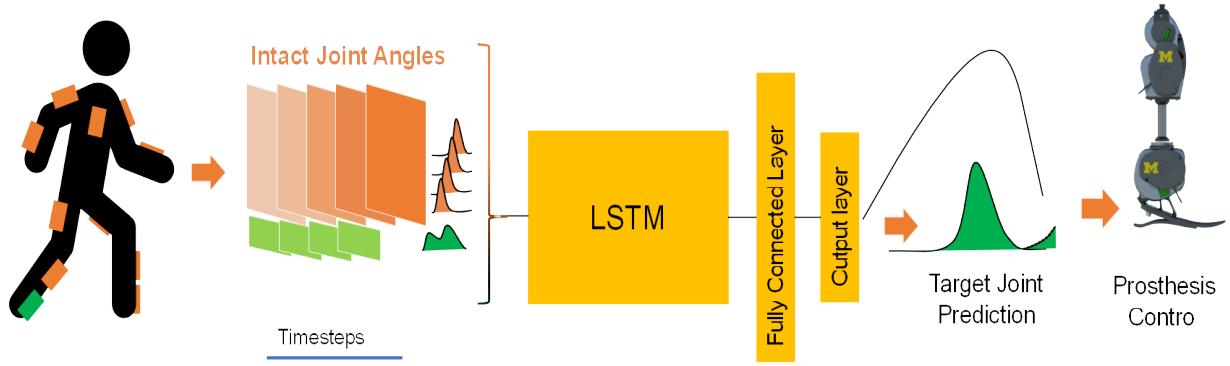


Figure 4.1: Neural Network Architecture. Each layer computes information based on the previous layer, but also using its own previous outputs and an internal memory.

Rolling Time Window During training, LSTMs backpropagate errors a specific number of time steps back. This parameter, known as the sequence length, affects the time scale that the LSTM cell state reasons about. Choosing a longer sequence length increases the number of parameters that need to be trained, increasing computational load and requiring more training data. Choosing a shorter sequence length increases the difficulty of learning time dependencies in the data. In practice, choosing a sequence length appropriate to the inherent temporal dynamics of the problem greatly simplifies training and performance of the network [83]. Training input samples were prepared as a overlapping rolling window of time series data of desired sequence length. The optimal sequence length was a hyperparamter we tuned for.

Loss Function and Neural Network Hyperparameter Optimization We used the mean squared error (MSE) between the predicted and measured joint angle as loss function to be optimized. This is common metric used for regression tasks in machine learning. Apart from the sequence length, the network also has several hyper-parameters that need to be optimized for different application domains.

Hyperparameter Optimization A combination of random and grid search was applied to optimize hyperparameters. Each batch was shuffled and random Gaussian noise was added to each sample to reduce over-fitting.

Optimized hyperparameters included batch size, number of epochs, number of layers (L), number of units in each layer (HU), the standard deviation of the injected noise, the regularization parameter for L2 loss (λ), and learning rate.

Hyperparameter	**Range/Values**	**Optimal**
Learning Rate	$[10^{-5} : 10^{-2}]$	10^{-3}
Batch Size	[1000,10000,50000,100000]	50000
Number of Epochs	[200,500,750,1000]	500
Input Sequence Length	[1,2,5,10,15,20,25,30]	10
Number of LSTM Layers	[2,4,8,12]	2
Number of Hidden Units	[4,8,16,32,64,128,256]	64
Regularization Rate	[0,0.05]	0
Random Noise (std)	[0.01 : 1]	0.02

Table 4.1: Hyperparameter values tested for optimal performance on Obstacle course dataset

Every 5 epochs, the performance of the model was evaluated on a validation set. The best performing model was saved and used to generate predictions and metrics on a test set. 30 trials were evaluated for each parameter set and the average RMSE was recorded. The optimal parameter value selection was based not just on the absolute best performance but also considering the overhead in time and computation needed to reach that performance. The range of parameter values tested is shown in Table 4.1. The optimal hyperparameter set was used to compare and evaluate performance.

Denormalization To report results in orginal scale, all predictions were denormalized using avarage minimum and maximum scaling factors extracted from training set only. This is common practice in machine learning as test set scaling factors are not known a priori.

4.2.7 Analysis

We use Root Mean Squared Error (RMSE) and the Pearson correlation coefficient (PCC) as our outcome measures to report performance for different activities, sensor groups and quantity of data. Ideally, the RMSE will be equal to zero degrees and PCC would be equal to 1.

Performance by activity

We compared performance of sagittal plane ankle and knee joint angle predictions for models trained on data from:

- Flat ground walking with random stops
- Stair ascent and descent with flat ground sections

- CHAMP tests

We combine all the activities in training and report errors for each activity evaluated separately. This prevents averaging out of errors across activities.

Dependence on sensor configuration

We assessed performance for flat ground, stairs, and CHAMP activities for the following sensor groups as inputs:

- Full body (20 joints) in all 3 anatomical planes
- Lower limb (6 joints) in all 3 anatomical planes

Dependence on data

We assessed performance for different amounts of training data by varying:

- Number of subjects included in training data.
- Percentage of data included from *every* subject in training data.

Significance and equivalence testing

R-package and Matlab were used for statistical analysis. Simple rhythmic movements of flatground walking should be easier to generalize than those involved in the CHAMP tests. Hence, we expected the performance to vary with activity. Similarly, more training examples collected from more subjects should allow the network to learn more variations and hence perform better. For these outcomes we use paired-sample t-tests. Effect sizes were calculated using cohen's d method [16].

In contrast, we expected most of the movement cues for flatground walking to be present in lower body. We also expect a diminishing return by adding more training examples from the same subject as they would likely be similar in nature. Hence, we hypothesize that sensor configuration and percentage of data from every subject will not result in significant differences in performance. For these outcomes, we perform equivalence testing. A relevant bound around the mean value is needed within which the results can be considered "equivalent". We apply the "two one-sided t-tests" (TOST procedure)[51], one for the lower bound and one for the upper bound, to show that the change is statistically equivalent to zero.

Upper and lower bounds of 3 degrees was used for flat ground activity. This value corresponds to the best case minimum detectable change (MDC) for sagittal plane ankle joint kinematics in literature[11, 102]. MDC for other activities has not been established. We discuss the motivation and implication of using this value in the discussion section.

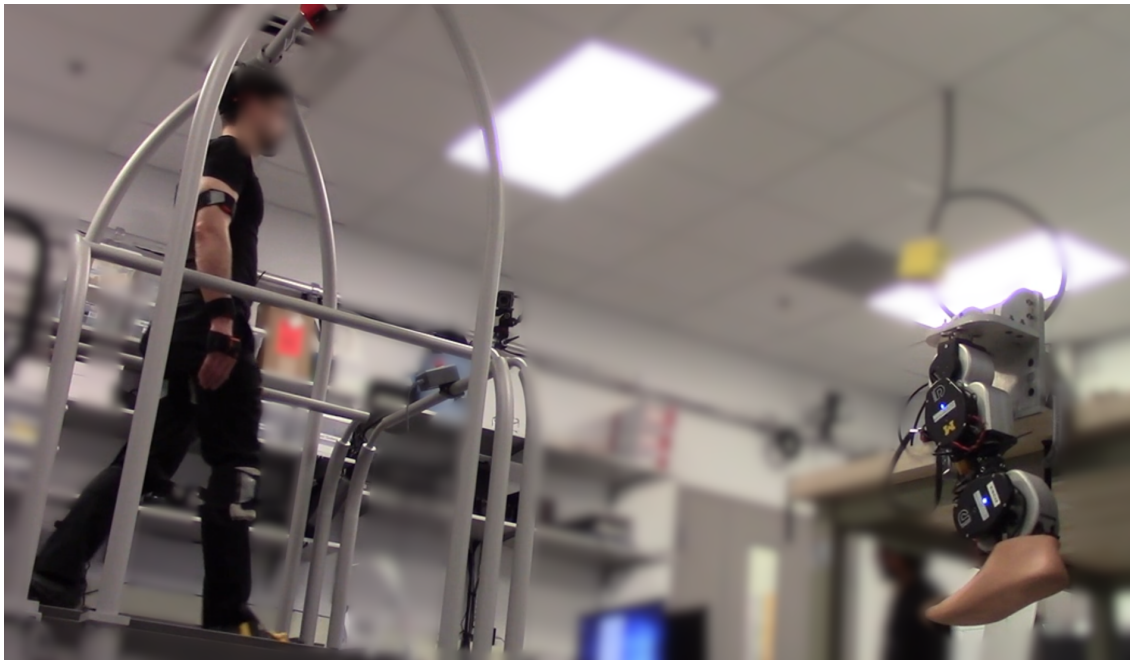


Figure 4.2: Benchtop setup with Open Source Leg (OSL). An individual wearing the motion capture sensors walked on the treadmill at self-selected speed. Live kinematics from the suit were used as inputs to a pre-trained network that generated right ankle and knee predictions. These predictions were used to actuate the OSL in real time.

4.2.8 Benchtop experiments on the Open Source Leg

A benchtop experiment was designed to test the performance of this control strategy on a powered leg prosthesis. The pipeline of the Coordinated Movement (CM) controller entails 1) acquisition of kinematics from Xsens, 2) pre-processing of the data (See Section 4.2.6 a,b), 3) predicting the prosthetic joint trajectories by the neural network, and finally 4) translating these predictions for lower-level control of the OSL. The lack of HD processing (See Section 4.2.5) of kinematics can introduce noise and error in predictions. The pre-processing of data and predictions are operations that take finite time. A delay or error in response of a load-bearing prosthesis could result in critical injuries in a real-life scenario. This benchtop test demonstrates the overall latency in the CM pipeline as well as the quality of live predictions and evaluates the safety of the CM controller prior to actual subject experiments.

We used the Open Source Leg (OSL) [10], a modular lightweight robotic leg designed to facilitate a common hardware test-bed for prosthetic control research. It provides an open source API to control position, torque and impedance of the knee and ankle joint making it an ideal platform to compare different control strategies.

A person with no amputation wearing an Xsens motion capture suit ambulated on a

treadmill at a self-selected speed (Fig 4.2). A real-time controller built using Xsens Python SDK acquired live kinematics as inputs. Right ankle and knee joints were omitted as they were the intended target prosthetic joints. An LSTM neural network (see Section 4.2.6) pretrained on offline flat-ground walking data (to match the treadmill walking) was used to predict right ankle and knee joint kinematics from the inputs. These predicted joint kinematics were encoded to actuator positions on the Open Source Leg mounted on a benchtop. The controller ran on a GPU powered laptop tethered to the OSL.

4.3 Results

The coordinated movement controller generated continuous real-time trajectory predictions for all the activities as shown in Fig. 3

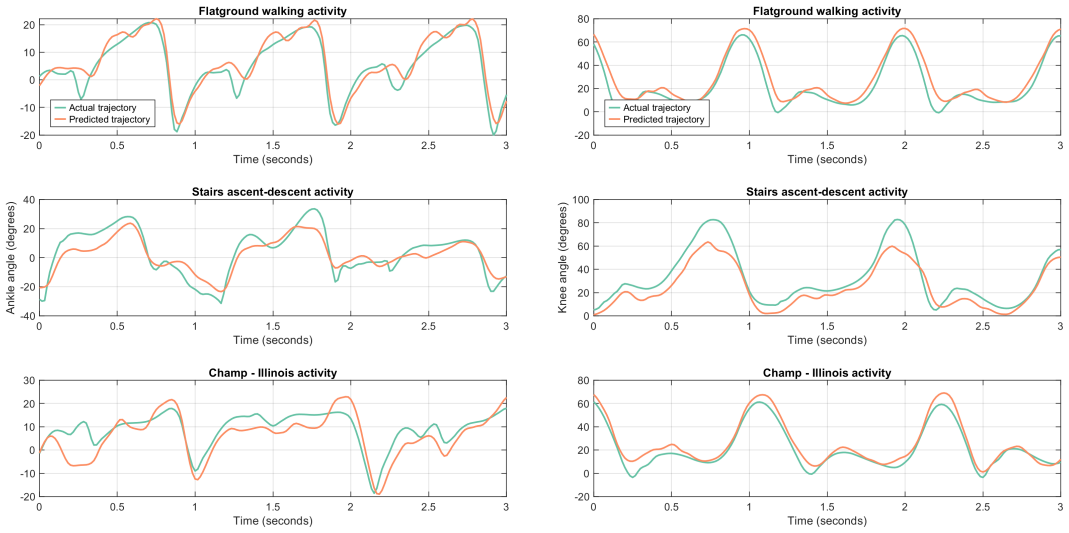


Figure 4.3: Ankle (left, positive y dorsiflexion) and knee (right, positive y flexion) joint predictions for 3 different activities generated by the same network. The trajectories shown for a test subject whose data was not part of the training data. About 3 seconds of actual measured (green) and predicted (red) trajectory for flat-ground (top), stair ascent-descent (middle) and Illinois Agility Test (bottom) activities are shown. Though these activities are presented separately, the network that generated these predictions was trained on a combination of all of them, and did not require activity categorization.

	Ankle		Knee	
	mean	**std**	**mean**	**std**
Flat-ground	0.8751	0.0162	0.9743	0.0049
Stairs	0.8951	0.0058	0.9598	0.0022
CHAMP	0.8318	0.0145	0.9191	0.0102

Table 4.2: Pearson correlation coefficients of predictions with respect to activities. Mean and standard deviations shown for ankle and knee joint sagittal plane predictions. Correlation dropped significantly for both joint predictions with increasing complexity of the activity.

	Ankle			Knee		
	Error (degs)	**ROM (degs)**	**%Error**	**Error (degs)**	**ROM (degs)**	**%Error**
Flat ground	4.99	50.31	**9.93**	3.99	77.15	**5.18**
Stairs	4.41	78.24	**5.6**	4.16	99.59	**4.19**
CHAMP	6.91	46.48	**14.8**	6.50	70.34	**9.25**

Table 4.3: RMS errors of predictions as percentages of range of motion. The CHAMP activity had the highest error percent for both joints.

4.3.1 Performance with respect to activities

Ankle and knee joint prediction performance varied with activity (Fig. 4.4) but were within 7 degrees RMS error and generally showed a high Pearson Correlation Coefficient (PCC) $>$ 0.85 (Table 4.2).

The CHAMP activity predictions had reduced performance for both joints, suggesting that the complexity of movements involved in an activity impacts prediction performance. PCC reflected a sharper contrast (Effect Size = 7.9) but RMS error showed a similar trend (Effect Size = 4.3). The RMS error of predictions was generally within 10% of the range of motion for each activity (Table 4.3). The only exception was the ankle joint predictions for CHAMP activities with about 14.8%. Similarly, the ankle joint predictions for CHAMP activities (Table 4.2) had the lowest correlation of 0.83. Predictions for stairs activity with training data from 40 subjects had approximately the same performance as the flat ground activity with 10 subjects. Training data from more subjects could have similar benefit to performance for CHAMP activity as we discuss in the following section.

4.3.2 Dependence on sensor configuration

All activities showed approximately the same performance ($<$ 0.5 degree change) when only the lower-limb sensor data was used as inputs. (Fig 4.6). Significant equivalence was de-

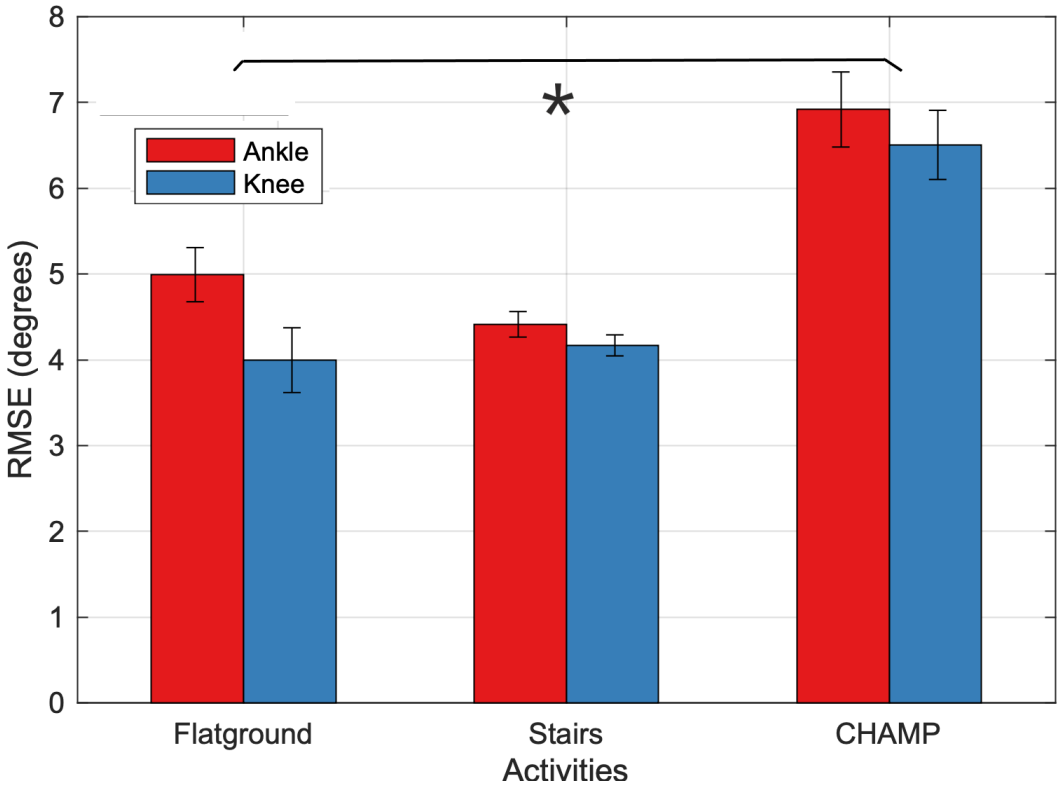


Figure 4.4: RMS error with respect to individual activities for ankle joint(red) and knee joint(blue) sagittal plane predictions. Performance was within 7 degrees RMS error for all activities and both joints. Complexity of activity significantly increased RMS Error. Asterisk indicates statistical significance

termined for flat-ground activity using the TOST procedure as described in Section 4.2.7. Equivalent performance with reduced set of sensors has beneficial implications for deployment with minimum instrumentation.

4.3.3 Dependence on data

Fig. 4.7 shows performance with varying amount of stair activity data. Prediction RMS error significantly decreased ($p<0.001$) with data from more subjects included in training the models (blue). This is an important result suggesting that performance of hard activities can be improved with training data from more subjects. Interestingly, the error remained approximately the same even when half the data from all subjects was not included in the training (orange). This could be due to the fact that adding more data from the same subject does not add any new variation to the overall repertoire of movements seen by the network.

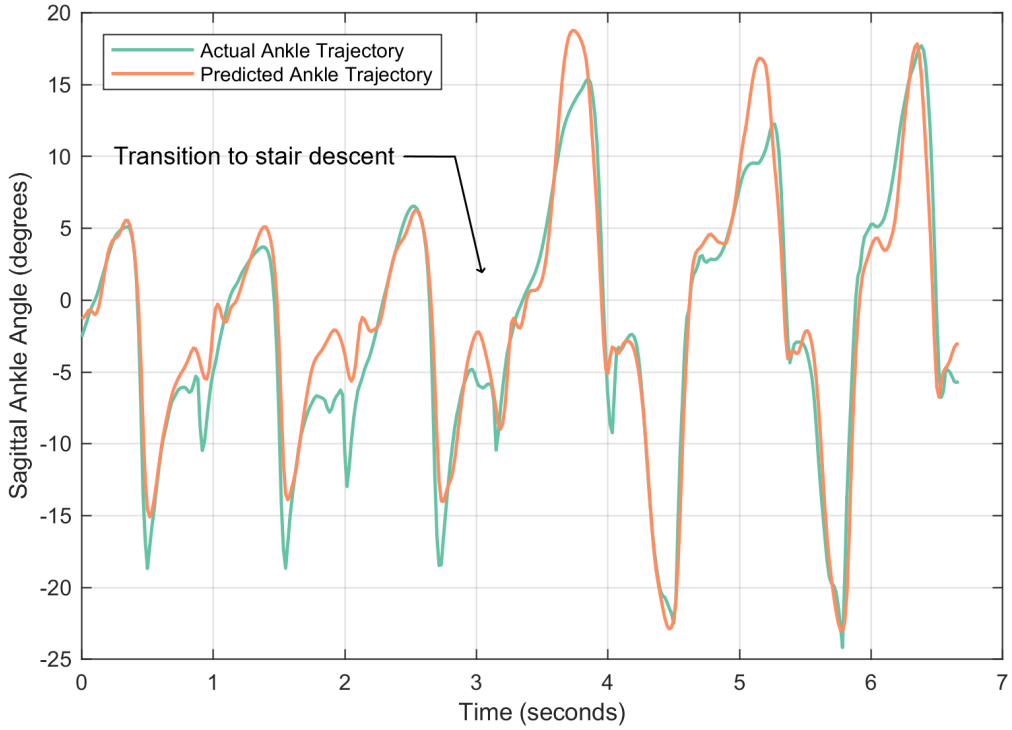


Figure 4.5: Ankle joint predictions showing continuous seamless transition from flat ground walking to stair descent.

4.3.4 Bench-top test

A single individual who was not a part of the training cohort was used for a bench-top systems test. Kinematics were collected as they walked on a treadmill at self selected pace. These data were played as inputs to predict ankle and knee joint trajectories for the Coordinated Movement controller for a prosthetic limb in real-time (Fig. 4.8). For the knee joint, the average RMS error was 5.62 degrees for the network predictions and 7.38 degrees for the actuated trajectory. These errors were higher than the offline results seen in Fig. 4.4 for flat ground activity. There was lag of 0.05 seconds in the response of the OSL. The error and the lag were calculated with respect to the actual knee trajectory measured on the test subject, which represents the ideal or the desired trajectory.

Discussion

The use of gait modes in previous prosthesis controllers has been a simplifying assumption and is not motivated with any physiological basis. An all purpose controller has not been

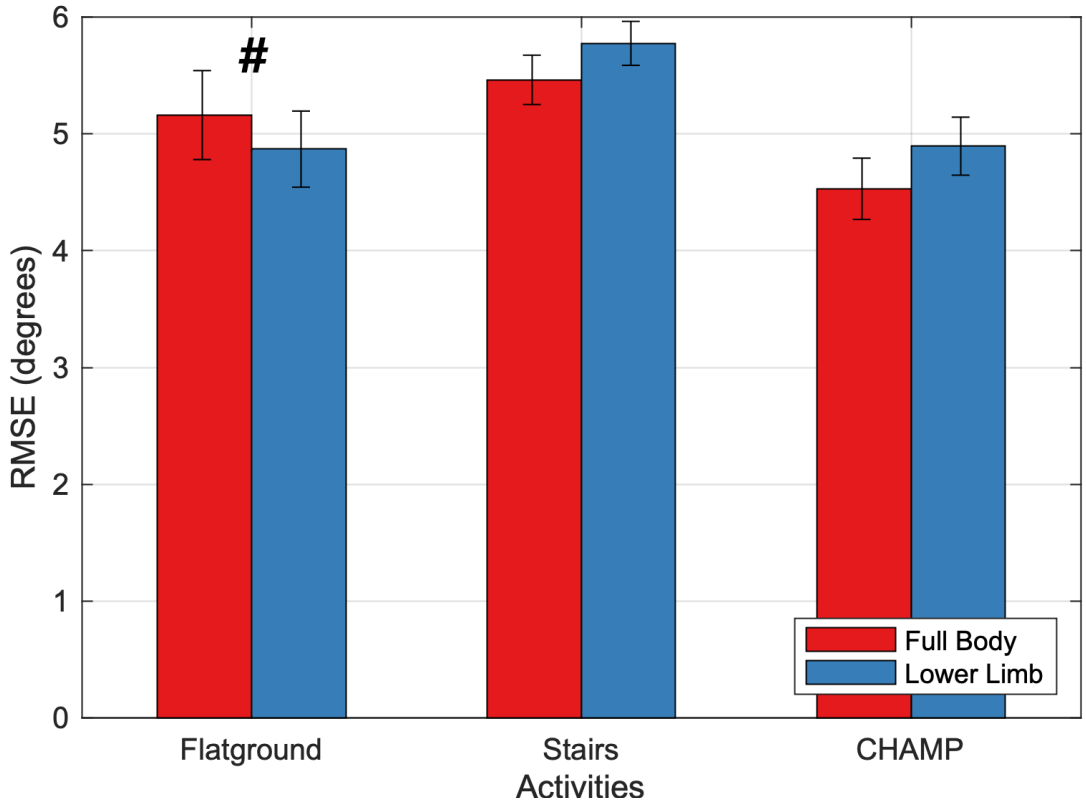


Figure 4.6: RMS error with respect to activities with fullbody (blue) and lower limb only(red) sensors as network inputs. Using only the lower limb sensors for training showed equivalent performance for flat ground activity.# indicates statistical equivalence

achievable. Similarly, powered prosthetic devices do not incorporate full body motion not because that information is useless, but because it has not been previously practical to measure in a camera instrumented gait analysis laboratory. Body-worn devices that track the user's motion continue to improve, smaller, and less intrusive, allowing us to capture unstructured movements outside the lab environments. In this manuscript we present a unified Coordinated Movement (CM) controller that leverages such a kinematic dataset towards continuous powered limb control.

A unified controller for varied activities

Our CM controller was trained on data from different activities and can generate movement for any of them, as well as the transitions between them, seamlessly. For example, Fig. 4.5 shows an example of ankle angle as a user transitioned from walking on flat ground to descending stairs. The kinematics of their contralateral limbs provided sufficient cues

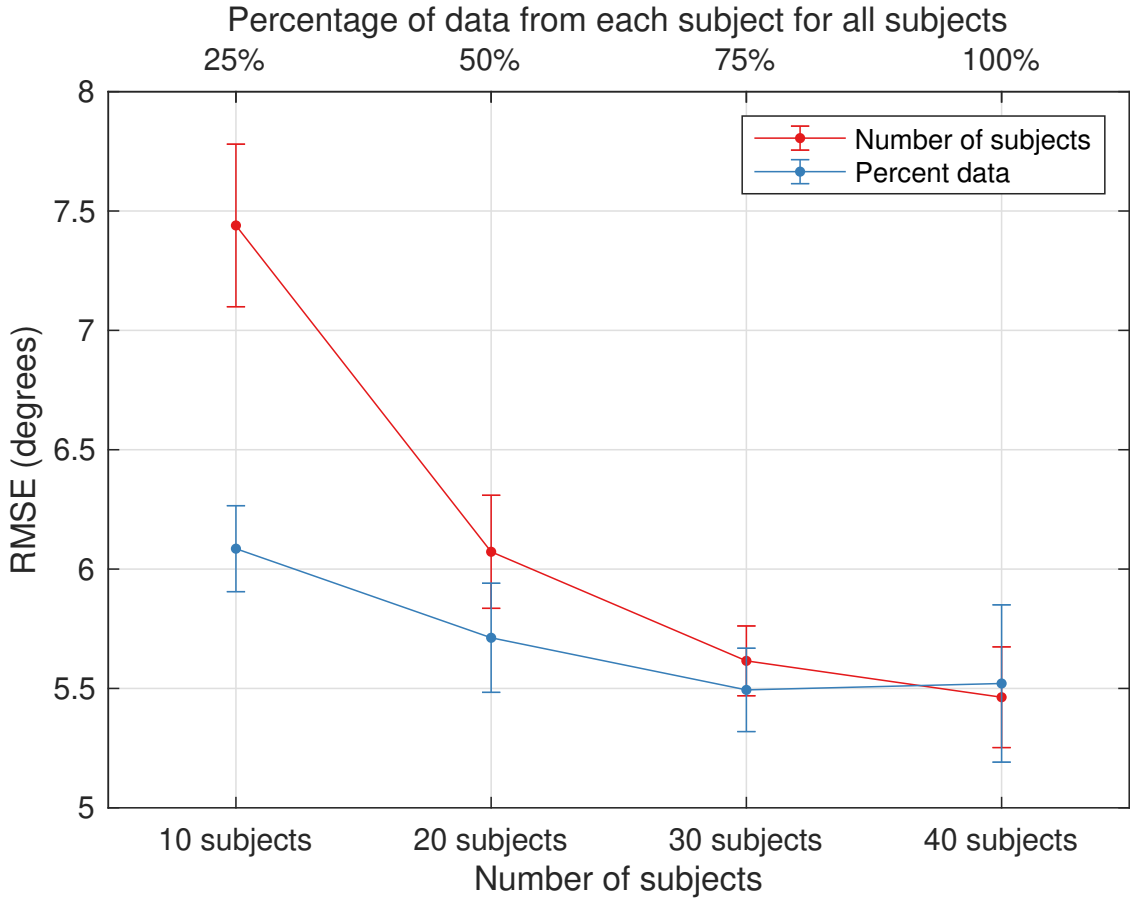


Figure 4.7: RMS error with respect to number of subjects (blue, bottom X-axis) and percentage of data used from all subjects (orange, top X-axis) for stair ascent and descent data. Performance significantly improved with more subjects included in training data. Keeping the total number of subjects the same (n=40), but using only 50% of the data showed approximately the same performance.

for appropriate ankle and knee angle predictions without explicitly categorizing ambulation mode.

Although the CM controller can generate predictions for all activities included in the training data, the performance for each activity varied. The CHAMP activity predictions (Fig.4.4) showed significantly higher RMS error than flat ground activity. This could be due to the complex collective of movements that included weaving around obstacles, walking sideways and backwards. This suggests that complexity of activity impacts performance and it seems likely that more training data could benefit such activities, as we discuss below.

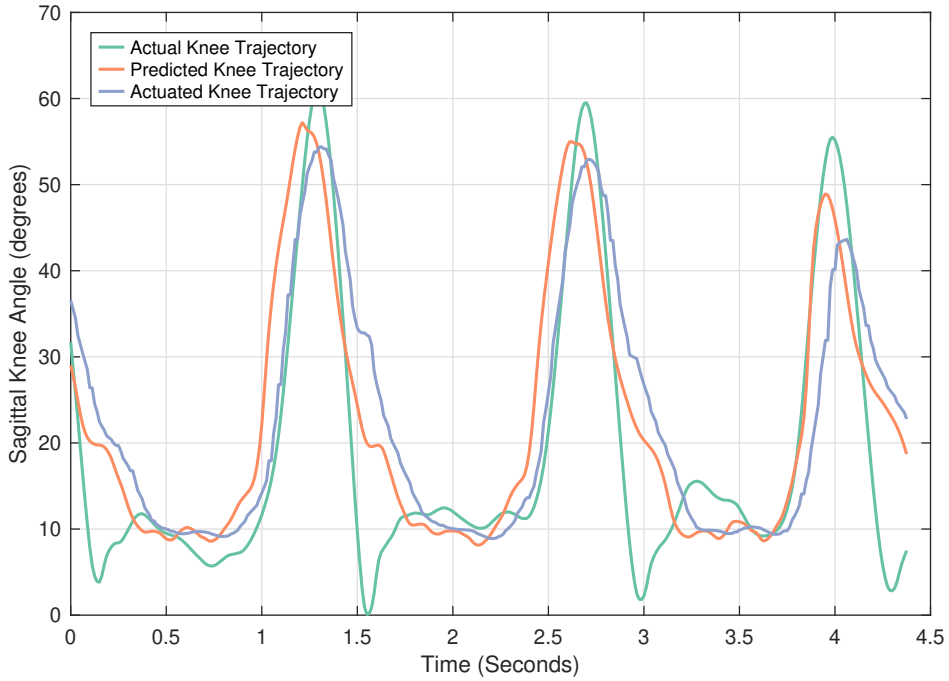


Figure 4.8: Knee joint predicted (orange) and actuated trajectories (blue) during the bench-top tests with treadmill walking activity. For this trial, predicted trajectory had an RMS error of 5.62 degrees and the actuated trajectory had an error of 7.38 degrees, with a lag of 0.05seconds with respect to the actual trajectory executed by the subject(green).

Predicting knee and ankle simultaneously Predicting both knee and ankle is more difficult than predicting only the ankle for two reasons. The first is simply that the network has two outputs to learn. The second is that the knee joint input data itself is highly salient for predicting the ankle joint. In our previous work [79, 80], we predicted only the ankle joint. The error we observe in the present study is higher, but only noticeably so in the CHAMP activity (Fig. 4.9). We speculate that simpler movements contain more redundant information in the remaining joints. For more dynamic and complex activities like CHAMP, the value of the ipsilateral knee joint is higher.

Sensor Configuration

An interesting result is that using only lower limb sensors had approximately the same performance as using full body sensors (Fig.4.6). This suggests that, in the interest of minimum instrumentation and cost, this system could be deployed with just lower-limb sensors without compromising overall performance.

We expected the CHAMP activity to benefit from the inclusion of the upper body sensors given the relatively complex movements involved. Evidently, most of the information for the

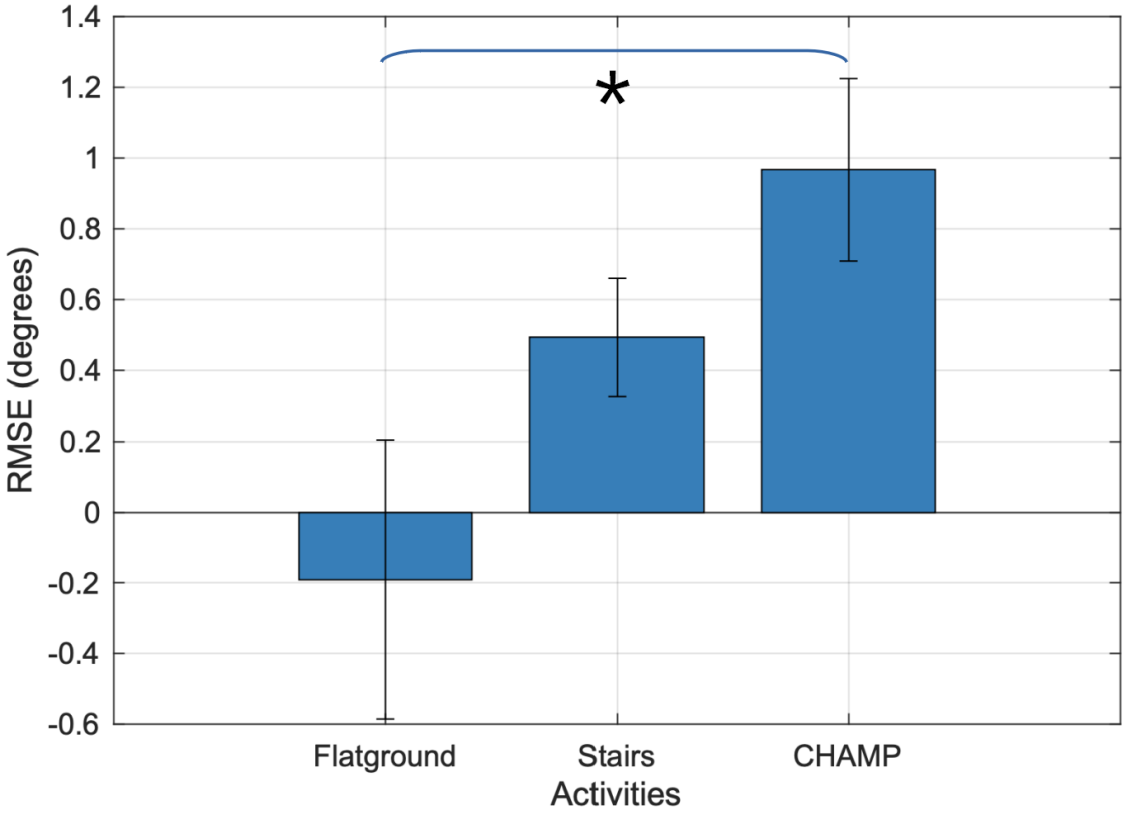


Figure 4.9: Change in degree RMS error for ankle joint predictions without the ipsilateral knee as one of the input joints. Cyclic activities like flat ground walking offer more redundancy in other joints whereas unique activities require more input joints for prediction. Asterisk indicates statistical significance

activities in this study was captured by the lower limb data. However, this result should be considered with a caveat. Upper limb movements are integral to maintain dynamic stability and perhaps more intricate activities that demand whole body co-ordination would benefit from full body data.

Performance Dependence on Data

The performance was significantly improved when more subjects were included in the training data. This peak performance was maintained even when half the data from *every* subject was excluded from training (Fig. 4.7). These results have two implications for more efficient data collection protocols in the future. Firstly, as expected, a data-driven approach relies on and benefits by including data from more subjects. Secondly, less data from more subjects

is better than more data from fewer subjects.

With equal number of subjects (n=10), the RMS error of predictions for stair activity were significantly greater error than flat ground activity. Inclusion of 30 more subjects (n=40) reduced this error by almost 2 degrees, bringing it closer to flat ground prediction performance. This suggests that relatively more training data are needed for more complex activities to achieve a performance similar to cyclic activities. We expect the result to hold for CHAMP activity as well.

Bench-top test

The bench-top test demonstrated the feasibility of our controller to actuate a prosthetic leg. The RMS error of the real-time predictions were slightly greater than those observed in offline analysis, underscoring the role of algorithmic data-processing in wearable motion capture. Raw motion-tracking data is inherently noisy. Xsens real-time engine mitigates this by accommodating sensor drifts and correlating independent sensor data to a human body model. A post processing engine further improves data quality by including past, present, and future samples. While offline analysis has the benefit of using the clean post processed data, the real-time control does not. This results in poorer prediction performance. This deviation in distribution of the data in test compared to training, known as covariate shift, is commonly seen in machine learning. This could be alleviated by using raw unprocessed data, or processed data with artificial noise, as training inputs to simulate the real-time data. Generation of synthetic training data using Generative Adversarial Networks (GAN) could allow the network to be more robust to sensor noise[39]. A more thorough long term solution could be to engineer an algorithmically light version of the post processing engine to operate on real-time data as well. The whole pipeline of operations resulted in a lag of less than 0.05 seconds in the response of the prosthetic leg. For upper-limb protheses a delay greater than 300ms is considered significant [18]. However, this value has not been established for lower-limb prosthes.

Limitations

Error bounds Our objective was to replicate normative joint trajectories for every instant in time. We chose our outcome measure to be the RMS error, commonly used for regression tasks. This makes it difficult to objectively compare with mode-based strategies which report accuracy in percentage of accurate mode classification. Moreover, it is unclear if the degree error reported in this study is within an acceptable threshold for practical use. For the case of flat ground walking, the RMS error is comparable to the Minimal Detectable Change (MDC) values of around 3 and 5 degrees[11, 102] for ankle and knee joints in sagittal plane.

For the lack of a better measure, we use an MDC of 3 degrees to determine equivalence in the case of flat-ground activity. Similar approach has been used in other studies[13] but to show a statistical difference due to intervention. Even though the 3 degree bound was the lowest value seen in literature, it is still large enough to show statistical equivalence for

all the activities in this study. However, statistical equivalence should not be confounded with practical equivalence. These MDC values correspond to gait measurement and may not equate to MDC for a load bearing prosthetic application. MDC for other activities and more importantly minimal clinically important difference (MCID) for gait and prosthetic control is critically lacking [3].

Use of normative trajectories We show that deep networks can represent and adapt to subjects *not* directly measured during training. So even though the prosthetic user's own gait may not have been recorded prior to amputation, this approach could still be employed. However, a major assumption with this approach is that normative trajectories collected from unimpaired subjects can be replayed for prosthesis users. Normative trajectories have been successfully used for prosthetic control[97, 84] and as tuning objective to optimize powered limb impedance parameters [101]. We expect similar performance with our controller. However, as prior studies have concluded, a more complex reference objective that accounts for variations of weight and dexterity of prosthetic limb could be necessary. The tuning of the lower level controller gains (PID) will most likely be needed to ensure a safe and comfortable ambulation.

Position control An impedance or torque based prosthetic controller is likely to be more comfortable. Current commercial wearable sensors provide only position estimates, limiting the target predictions in this study to joint kinematics only. However, to replicate force plate data in outside laboratory conditions there is an increased interest in estimating joint kinetics using wearable sensors[7, 59]. Upon availability of such reference data, this data driven methodology can be used to generate joint torque predictions as well.

Improvements and Future Work

Experiments in preparation are geared towards assessing the performance of this control strategy on human subjects as well as to compare it to other mode-based control strategies.

Non-time varying and subject specific features contain rich contextual information and have shown to improve network accuracy in bio-medical applications [20]. Including non temporal subject-specific data that affect gait kinematics, such as gender [14], body dimensions [12], age[66] etc. could possibly improve performance.

Conclusion

This study aims to address the challenges faced by powered lower limb users during unstructured activities such as side shuffling and weaving around obstacles. We demonstrate here that a data driven approach could be applicable to realize continuous control of powered prosthesis without explicit categorization of such movements.

65 subjects wore a motion capture suit and performed various rhythmic and arrhythmic activities. A recurrent neural network was trained to predict ankle and knee kinematics using the remaining joint kinematics as inputs. Performance was within 7 degrees RMS error for test subjects excluded from training examples. These errors are generally less than 10 % of the ROM of the corresponding activities.

A pilot benchtop study was conducted, demonstrating that the predictions and controller can all be run in real time on real hardware with comparable performance.

Chapter 5

AIM 3A: UNSUPERVISED LABELING FOR LOCOMOTION MODE CLASSIFICATION

Increasing the repertoire of movements via the creation of new mode classes entails associating an input sensor pattern with the corresponding mode label. A significant challenge in this process is labeling the training data. This process is resource-intensive, especially for vision data as each image needs a label. In this chapter, we describe our Primary Aim 3a, an unsupervised method to label training data. In the subsequent chapter 6 we elaborate this method to train a vision-to-mode classifier by using the generated labels as targets.

The method involves clustering kinematically similar knee gait cycles using an unsupervised machine learning model. These cluster labels constitute "modes," by virtue of their similarity in the data. The unsupervised nature of the clustering algorithm facilitates regularities present in natural movements dictate the number and the characteristics of mode, eliminating subjective bias while categorizing data.

5.1 Introduction

The locomotion mode-based lower-limb prostheses control strategy adapts to the large variety of terrain types by categorizing them into a handful of mode classes. Each mode is designed to generate a pre-defined reference trajectory suitable for navigating a particular type of terrain (stair ascent, ramp). Machine learning and pattern recognition of sensor data are employed to select the 'right' mode for any given moment.

Mode-based control has the advantage of being more predictable and tractable, at least when the desired behaviors lie within the modes. To increase the vocabulary of this strategy

Figure 5.1: Very different visual scenes can have similar kinematic behavior. For example both scenes above require flatground walking.

to encompass more movements requires training classifiers to associate sensor data to the right mode class. Regardless of the input sensor modality, this process requires a large amount of training data labeled with appropriate target mode class. Current labeling is done manually [62, 114] which is time and resource consuming, even for a modest number of modes. In the case of vision as the input sensor, this process can be intensive as 2 visually distinct scenes can have the same desired movements (Fig. 5.1). For example, in [52], 37,000 vision frames were manually labeled by the researcher for data from a single subject with 3 mode classes. Manual labeling can also introduce subjective bias as the clinician or researcher decides the corresponding mode label[62, 52]. This has often resulted in the generalization of movements present in the data to broad predetermined mode categories. This coarse categorization might ignore the subtle deviations from the normative mode behavior present in the data that merits a new mode class.

Unsupervised labeling of training data for mode-based control strategy can have the following benefits:

1. Remove human bias and allow unique and natural movements present in training data to be identified and categorized into separate mode classes.

2. Free up researcher/clinician time to facilitate more data collection to capture other interesting and desired movements.

The researcher and the clinician are still integral in a supervisory role for the final selection of desired movements and the valid number of modes, as we shall describe below.

5.1.1 Unsupervised Labeling using Cluster Analysis

Our third primary aim addresses the need for a richer repertoire of movements while still retaining the benefits and tractability of a mode-based strategy by using an unsupervised learning approach. Unsupervised learning is a type of machine learning algorithm used to draw inferences from datasets consisting of input data without labeled responses. Specifically, we employ a technique known as *cluster analysis*.

Cluster analysis is a statistical technique that provides an objective, quantitative classification system by separating individuals into homogenous groups based on selected input parameters. It has been successfully applied to identify patterns of gait deviations in children with cerebral palsy [92], to distinguish the walking parameters of young from elderly subjects [100], to segment gait phases [77]. The clusters are modeled using a measure of similarity defined upon metrics such as Euclidean or cosine distance. Lack of 'correct' labels or responses makes objective validation of cluster results challenging. However, qualitative and quantitative analysis for resulting clusters in previous studies shows promising results. We use similar metrics for validating the results in this study.

We apply cluster analysis to gait cycles in the training data to derive terrain mode labels for each cycle. In prior studies, lower-limb joints as input features have consistently shown

the most predictive power [92]. Based on this, we use time-normalized knee gait cycles as input features. Kinematically similar knee trajectories reflecting similar activity or terrain types would be clustered together. Hence, the cluster group labels constitute "modes," by virtue of their similarity in the data. The parameter to select the number of cluster groups, and hence the kinematic patterns extracted up by the clustering algorithm, can be arbitrarily determined by the clinician. The grouping itself is determined by natural movements in the data. We demonstrate how the algorithm can enable making an informed decision and examine the kinematic patterns extracted by varying this parameter.

Cluster analysis models cannot assign labels to new data. To evaluate and predict on an unseen test data, we train a K-Nearest Neighbor using the labels generated by the clustering algorithm. We describe the method to train and evaluate the cluster analysis model and the KNN classifier in the following section.

5.2 Methods

5.2.1 Experiments and Data Collection

Ambulation data was collected for a total of 10 healthy participants with no amputation or other mobility impairments. Recruitment and human subject protocols were performed in accordance with the local University of Washington Institutional Review Board approval and each subject provided informed consent. De-identified data can be made available, via a data use agreement, upon request to the authors.

The subject's anthropometric details were recorded and 17 wearable sensors part of the Xsens Motion Capture system were placed on their body. This was followed by a system specified calibration procedure and a brief test to see the quality of data being recorded. The subjects then performed 10-15 minute trials of the desired activity. The subjects were instructed to walk naturally at a self-selected pace. The data for these activities was collected in public spaces during active business hours, with the intent that normal gait dynamics and corrections would appear in the example data.

Each trial started with participants performing flatground walking in a cluttered classroom environment with chairs, followed by sections of flatground walking in open corridors. The participants were then instructed to ascend a flight of stairs to reach the next level, which involved a flatground walking in corridors. Participants returned to the original level by descending the flight of stairs. There were brief sections of atypical movements such as opening the classroom door to enter the corridor section, short sections (2-3 steps) of flatground in between flight of stairs.

5.2.2 Instrumentation

We collected locomotion data using the Xsens Awinda suit [4], consisting of 17 body-worn sensors placed at key locations. Each sensor has a tri-axial gyroscope, accelerometer, magnetometer, and barometer. Xsens Analyse software integrates these individual sensors and

renders a full-body avatar. After a system specified calibration, the software provides position and joint kinematics in a 3D environment. Although other data such as limb-segment position, orientation, acceleration are available, we used only joint angles for this study. All angles are in 1x3 Euler representation of the joint angle vector (x, y, z) in degrees, calculated using the Euler sequence ZXY using the International Society of Biomechanics standard joint angle coordinate system [105]. Data, sampled at 60 Hz, from a total of 22 joints in 3 anatomical planes (sagittal, frontal, transverse) were captured for each trial, which results in 66 total possible features for our machine learning methods.

5.2.3 Data-processing

The input to our clustering models was the right knee joint kinematics. Segmented gait cycles were temporally normalized to 100 percent. Each gait cycle was considered a single sample, in total yielding 6766 gait cycles in the training data. Relative euclidean distance between each sample was used to measure similarity for the clustering process.

Due to the lack of insole data, knee gait cycles were segmented by using the contralateral knee peak flexion as cutoffs (Fig.5.2). Similar methods have been used in other studies to segment gait data lacking force plate data. This method is not ideal for gait segmentation. However, this method relies on relative distances between knee gait cycles. The consistency of using the same strategy for segmentation across the training should yield similar results compared to methods like using foot insole. MATLAB command *findpeaks* was used for determining the peaks.

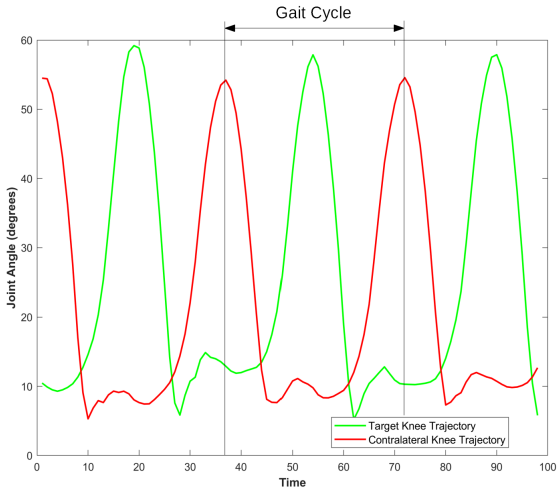


Figure 5.2: Knee gait cycles were segmented using the peak flexion angle of the contralateral knee joint as beginning and end of the cycle. Although an imperfect method, the consistency of the methodology should yield the same results as using foot insole based gait segmentation

5.2.4 Machine Learning Models

We use unsupervised learning to group gait cycles in the training data, which yields mode labels as responses. Some of the clustering algorithms (kmeans, kmedians etc.) can label new instances based on the model created. The hierarchical clustering models operated by agglomerating individual samples and do not partition input space. To classify a new instance, a classifier has to be trained using the labeled responses generated by the clustering model. This is the traditional supervised learning approach.

Unsupervised Machine Learning - Cluster Models

Clustering algorithms can be broadly classified as hierarchical methods or non-hierarchical methods such as K-Means. A hierarchical procedure in cluster analysis is characterized by the development of a tree-like structure known as a dendrogram. Non-hierarchical methods are faster to use than hierarchical ones but do not allow step-by-step inspection of the clustering process. Hierarchical methods like Agglomerative Clustering (HAC) or Density-based spatial clustering of applications with noise (DBSCAN) however provide better interpretability even though more advanced supervised techniques such as artificial neural networks could marginally improve classification accuracy [77]. In this study, we compare Agglomerative and KMedoid clustering methods. We evaluate performance using computational metrics and visual analysis.

Supervised Learning - KNN classifier

Hierarchical Clustering models cannot be applied to predict on new data. A KNN classifier was trained to predict the mode classes of an unseen test set. The inputs to the KNN classifier consisted of knee gait cycles from training datasets with labels generated by HAC as targets. In KNN classification, a sample is classified by a plurality vote of its neighbors, with the sample being assigned to the class most common among its k nearest neighbors. We report the classification accuracy of test set.

All machine learning algorithms were implemented on Python using the scikit-learn machine learning library [73]

5.2.5 Performance Evaluation

Validation of the resulting gait clusters is an important step in the clustering process. The lack of true labels in unsupervised clustering makes objective evaluation of clustering performance difficult. Halkidi et al[30] described three criteria to examine cluster validity

1. External criteria evaluated cluster results based on some externally known results, such as externally provided class labels.

2. Internal criteria quantify the data clustered itself (internal information) without reference to external information. It can be also used for estimating the number of clusters and the appropriate clustering algorithm without any external data.

3. Relative criteria evaluate by varying different parameters for the same algorithm (e.g. changing the number of clusters).

R-ratio is used by prior studies as a computational metric to determine the optimal number of clusters [15, 98]. This is a measure of the reduction of the within-cluster variability.

$$R = \left[\frac{e(N,K)}{e(N,K+1)} - 1\right](N-K+1)$$

where e(N, K) is defined as the summation of within-cluster square distances for N patterns and K clusters. The algorithm is stopped when the R ratio is large corresponding to a large reduction in the within-cluster variability. This indicates that the clusters are quite homogeneous

Silhouette score, Dunn-Index, and coefficient of variation are some common metrics to quantify within-class proximity and inter-class separation for internal validation. ANOVA comparison is used to quantitatively verify that variables of interest, e.g. knee kinematics, in clusters are widely separated and distinct. Studies generally use a combination of these, although final visual inspection is generally considered crucial[92].

We evaluate the cluster models as follows -

Optimal Number of Clusters and Verification: We use R Ratio and within-cluster sum of squared error as a means to select the optimal number of cluster groups in the training data (relative criteria). We use the Silhouette score to compare different clustering algorithms(internal criteria).

For external criteria, 300 random samples in the training data set were manually labeled to belong to one of the 3 dominant classes - flatground, stair ascent, and stair descent. We report the percentage accuracy of these samples being assigned the right cluster by the HAC algorithm

Kinematic Analysis: ANOVA and Posthoc Tukey We analyze knee kinematics extracted from K=3 and K=4 number of cluster groups. We perform ANOVA on the group samples followed by the post-hoc Tukey test to show differences between groups. Group mean and 25 percentile SD of the trajectory for each cluster group is shown for visual analysis.

Unseen Test Data : Kinematic data from a test subject excluded from the training cohort is manually labeled to serve as group truth for performance evaluation of the KNN classifier. We report the percentage of accurately labeled gait cycles.

5.3 Results

In this section, we report the results from the unsupervised labeling of training data using the Hierarchical Agglomerative Clustering (HAC). We determined the optimal number of clusters to be K=3, which encompassed flatground, stair ascent, and stair descent modes. The dendrogram of the HAC provides a visual representation of the cluster groups. We explore the cluster groups and the kinematic patterns when the parameter is changed to K=4. The performance of the clustering algorithm was evaluated on a subset of the training set. A KNN classifier trained on the same training inputs had an accuracy of 94% accuracy for an unseen test subject data.

Optimal Number of Clusters and Comparison of Cluster Models

R ratio peaked at K=3 number of clusters indicating 3 dominant kinematic patterns (Fig 5.3a. The silhouette score also peaked at K=3. Both clustering algorithms had approximately similar performance both metric (R-ratio and silhouette score)

Dendrogram

A dendrogram aids in visualizing and interpreting the resulting cluster groups along with their sample sizes, their mutual similarity by clustering metric, or linkage distance. Going from the top, the dendrogram allows informed decisions about the similarity of the occurring patterns or cluster groups in the data. Leaves represent the clusters. Leaves in the same branch are similar in terms of linkage distance between the clusters.

For eg. Fig. 5.4 shows the last 4 cluster agglomerations that resulted from 4 different kinematic patterns existing in the training data set. Further analysis of the samples in each of the cluster groups revealed that the groups correspond to flatground walking around chairs as obstacles, flatground walking in long corridors, stair ascent, and stair descent.

The dendrogram demonstrates the ability of the HAC to differentiate between flat ground walking in long corridor sections versus the flat ground walking inside classrooms. The latter which involved avoiding chairs could be considered a different ‘mode’ for prosthetic control. When this level of distinction is not desired, the group will be merged into a single ’flatground’ group, still distinct from stair ascent and descent.

Kinematic Patterns in Cluster Groups

The parameter K, the number of clusters decides the truncation of the clustering tree. This decision can be based on prior knowledge of the terrain types in the training data or by analyzing the dendrogram (Fig5.4 and the resultant cluster groups. For e.g. with K=3, the kinematic patterns extracted are shown in Fig 5.5a. These patterns correspond to flatground, stair-ascent and stair descent.

If the number of overall classes was selected to be 4, a fourth cluster emerges with a pattern resembling flatground walking but with subtle variation (Fig 5.5b). This variation

(a) R-ratio for the two clustering algorithms

(b) Silhouette score for the two clustering algorithms

Figure 5.3: Hierarchical Agglomerative Clustering (HAC) and K-Medoid had approximately similar performance for R-ratio (a) and Silhouette score (b). Both metrics (higher value is better) favored the optimal number of clusters to be 3 for both clustering algorithms.

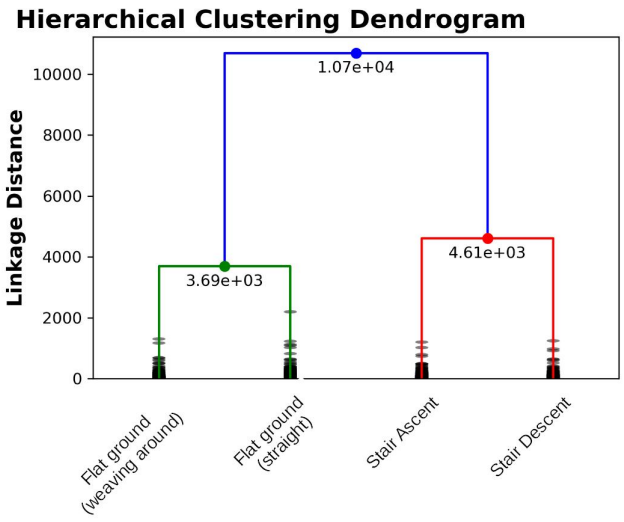


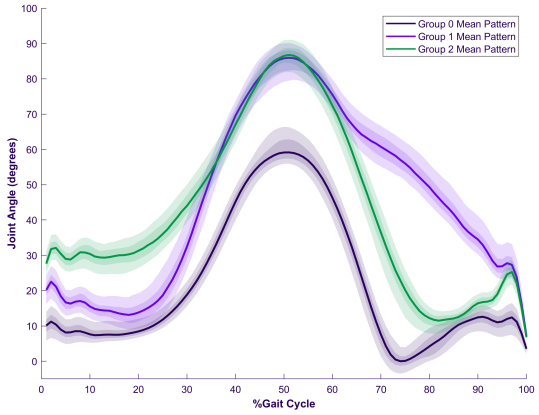
Figure 5.4: The dendrogram is a graphical representation of the cluster groups and their relationships. For e.g., with group labels starting with 0 from the left, cluster groups 0 and 1 belong to the same branch. Cluster groups 2 and 3 are distinct from groups 0 and 1. Visual analysis of the patterns in each of the groups reveals that groups 0 and 1 correspond to flatground in two different scenarios. Group 2 cluster contains all the stair ascent samples and group 3 was determined to contain stair descent samples.

corresponded to flat ground walking inside classrooms, avoiding the chairs as obstacles. Depending on the choice of the researcher/clinician, this pattern could be categorized as a separate class or lumped under a single flatground class, as is the case when K=3.

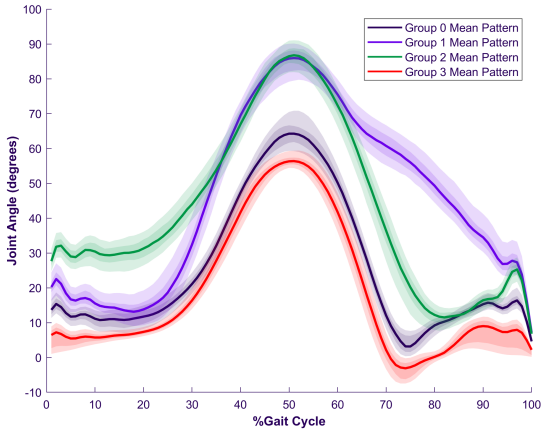
The average pattern for each group could be used as the reference trajectory for mode-based control.

Clustering Performance

300 random samples in the training data were manually labeled to belong to one of the 3 dominant classes - flatground, stair ascent, and stair descent. The cluster labels of these samples were analyzed to verify if the algorithm accurately assigned these samples to the right cluster groups. The confusion matrix is shown in Table 5.1. HAC had an overall accuracy of 95%, with 97% accuracy in grouping flatground samples, 100% for stair ascent samples, and 94% for the stair descent samples.



(a) Extracted patterns when the number of cluster groups is chosen to be 3



(b) Extracted patterns when the number of cluster groups is chosen to be 4. Group 0 and 1 correspond to flatground walking with and without obstacles. They are grouped together when K=3 (top) but can be separated into two separate modes when precise control is desired.

Figure 5.5: Mean patterns and 25% percentile of samples when the number of clusters is chosen to be K=3 (top) and K=4 (bottom). A new variant of flatground walking is extracted when K=4 corresponding to flat ground walking avoiding big obstacles (chairs). The mean patterns can be used as reference trajectory for the corresponding mode

Actual \ Predicted	Flatground	Stair Ascent	Stair Descent
Flat ground	256	0	2
Stair Ascent	6	18	0
Stair Descent	1	0	33

Table 5.1: Confusion matrix of actual and predicted cluster group labels for the 300 randomly selected samples. Samples were manually labeled to ascertain actual grouping.

Test set Prediction Performance

The generated cluster groups with their labels were used as targets to train a KNN classifier with the intent to classify the kinematic knee gait cycles of an unseen test subject data. Ground truth information was determined by manual labeling.

The classifier accurately predicted classes of 94% of gait cycles. These class labels will be the basis of locomotion mode class. Interestingly, genuine variations in gait from the three main classes were detected by the model but were mislabelled. For e.g., the transition to stair ascent was detected to be different from flatground and stair ascent gait. But due to lack of the right class to associate this pattern with, it was labeled as stair descent. Similarly, the flat ground sections between the flight of stairs were accurately distinguished but were classified as stair descent. For simplicity, ground truth for these deviations was assigned as one of the three dominant classes neighboring the section. A higher number of clusters could result in these aberrations being clustered into separate mode classes.

5.4 Discussion

In this chapter, we address the need to increase the repertoire of movements in the mode-based control strategy to encompass more diverse activities. Previously, this would entail the resource-intensive process of manually labeling the sensor data for mode recognition. This process is also prone to subjective bias as the researcher categorizes data with the corresponding pre-determined mode label. We demonstrate a method to implement this process as an unsupervised learning problem requiring minimal human intervention. The categorization of data is dictated by the uniqueness of the movements thereby eliminating bias.

The data in this study comprised of 3 very unique 'modes' along with some atypical movements. To broaden the repertoire of movement beyond the current mode-based capability would require more diverse terrain types. However, the aim of this study is to demonstrate the capability of the method to categorize movements without human bias. For e.g. the flatground activity in the classroom environment involved slight deviations relative to the open corridor environment. Upon further analysis, we observed these deviations were due to

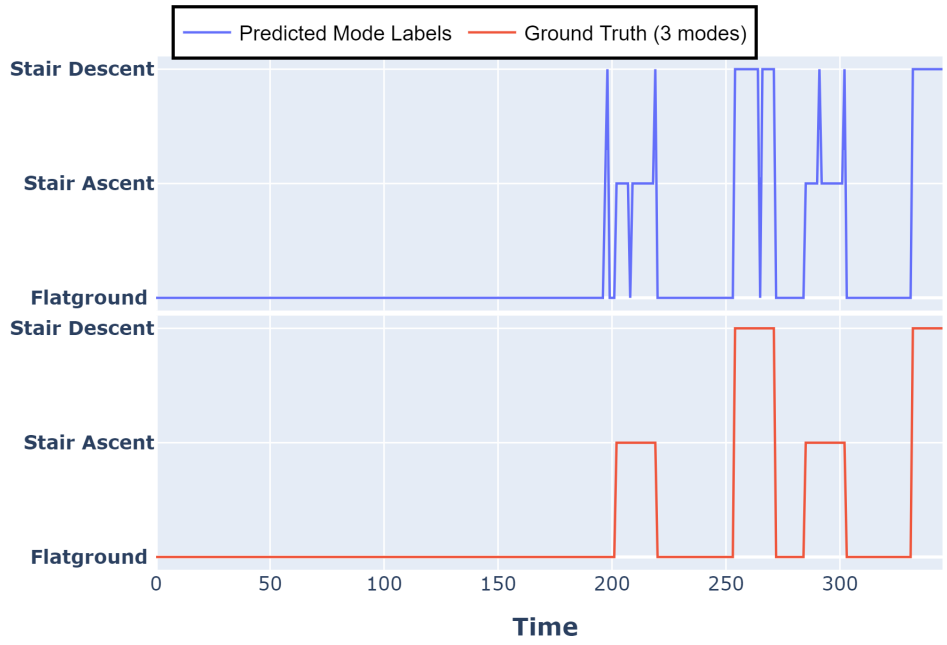


Figure 5.6: A KNN classifier was trained using the labels generated by the clustering model. Classification of gait cycles of an unseen test subject data had a mode classification accuracy of 94%. Ground truth was assigned to be one of the three dominant classes (flatgound, stair ascent and descent). Genuine variations in gait were extracted by the model but were mislabelled. For e.g., the transition to stair ascent was detected to be different from flatground and stair ascent gait. But due to lack of the right class to associate this pattern with, it was labeled as stair descent.

the weaving around the obstacle(chairs). Human labeling would generalize these into a single flat ground mode despite their slight differences. The clustering algorithm could differentiate these enough to form their mode classes.

We use knee gait cycles as input features to cluster gait data due to its high correlation to environment features. Other lower body joints have also been used towards clustering gait[92]. Full body or other joints as input features could improve clustering performance.

The gait cycles in this study were segmented using the peak contralateral knee flexion. This would lead to slight inaccuracy. However, the method we apply relies on comparison of knee cycles, which presumably remains unaffected by choosing different points in the gait phase to segment cycles.

The similarity was measured using Euclidean distance as the metric. Other distance metrics such as Dynamic Time Warp distance have been used in studies towards measuring the similarity of knee gait cycles[115, 76]. Future studies will focus on a thorough investigation of other distance metrics and clustering algorithms.

5.5 Conclusion

Limited number of movements is a major impediment towards usage of powered limbs. A significant challenge in expanding the number of movements in current prosthetic control strategy has been the need for labelled training examples. In this chapter we demonstrate a method for labeling training data without human intervention. In the subsequent chapter we use these generated labels to train a vision classifier to explicitly sense environment and anticipate terrain changes.

Chapter 6

AIM 3B VISION FOR PROSTHESIS CONTROL USING UNSUPERVISED LABELING OF TARGET MODES

Transitioning from one activity to another is one of the key challenges of prosthetic control that we address in this dissertation. Vision sensors can be a window into the upcoming activities of the user providing almost 2 secs of look-ahead relative to body sensors (EMG, mechanical). This look-ahead window, also known as lead-time, could be employed to anticipate and trigger transitions and provide a smooth user experience.

A significant bottleneck in using vision sensors has been providing labeled training examples. In the previous chapter, we discussed an unsupervised method to acquire mode labels for kinematic gait cycles in training data. In this chapter, we correlate the acquired labels with images from the same training data to train a vision classifier. The classifier predicts the target mode an average of 1.8 seconds before the kinematics changes. We report 96.6% overall and 99.5% steady-state mode classification accuracy. Apart from using an unsupervised model generated target labels, this study differs from prior work in applying transfer learning for the vision classification. Transfer learning utilizes the learned knowledge from a source task to a different task. This improves generalization, especially when the target task has a limited training dataset, as is often the case with prosthesis studies.

6.1 Introduction

Powered prostheses are constantly improving, becoming lighter, more rugged, and more capable with each iteration [10, 55]. Designing effective control systems for these advanced robotic devices to emulate human locomotion posits a challenge. Most current prosthetic control architectures can generally be described as a 3-layer hierarchical Finite State Machine [96, 113]. Intent recognition or estimation of the user's locomotive intent is performed as the high level. The mid level translates this intent to generate a corresponding biomechanical reference trajectory. The low level comprises a device-specific controller responsible for tracking the trajectory.

High-level intent recognition is based on modes of operation corresponding to activities, such as flat-ground walking, running, or stair ascent. Machine learning techniques are used for pattern recognition of sensor data to estimate the current mode and transitions. Training the machine learning models require examples of sensor pattern labeled with corresponding target modes.

Most research and commercial powered prosthesis use body-based sensors such as EMG and mechanical sensors for intent mode recognition. They focus more on user intent and are

Authors	*Sensor Type*	*M/L Model*	*Ground Truth Basis*	*Number of Modes*	*Results*
Massalinet al	Depth	SVM	Foot placement	3	94% OCA, -0.33 ms(delayed)
Zhang et al	Depth	CNN	Foot placement	5	98% OCA, + 3.3s lead-time
Yan et al	Depth	FSM	Foot placement	4	100% (SS) + 82% OSE
Huang et al	RGB	Bayesian NN	Foot placement	3 types of flat + 2	95% OCA, 89%(for 1s lead time)
Nili et al	Depth		Visual field	2	98% OCA
Brock et al	RGB	CNN	Visual field	3	94% OCA
Torres et al	RGB	CNN	Visual field	2	98% OCA

Table 6.1: A summary of studies using vision for prosthesis control. Performance reported as Overall Classification Accuracy (OCA) or Steady-state Accuracy (SSA), and lead times during transitions

considered implicit environment sensing modalities as they indirectly sense the environment by measuring user and device state.

Lack of explicit environment understanding results in poor control performance more evident during terrain transitions [40]. EMG activity precedes movement by about 100ms [8]. This precludes its usage to detect environment and predict locomotion modes of amputees in advance. They are also user-dependent sensors requiring subject-specific calibration and training. User independent training could improve classification accuracy [109], and presumably allow off-the-shelf use of the device. Implicit sensing also compromises the fault tolerance of the system by focusing solely on user state for error correction [32].

A user-independent sensor with ability to directly sense the environment, and anticipate transitions in advance would allow for triggering a change in controls at the right moment. It would also improve overall robustness of the system [96, 95].

6.1.1 Vision for Explicitly Sensing Environment

Humans use vision to sense the environment and fluidly adapt to upcoming changes. Even with wide variability of terrains, humans consistently looked 1.5 seconds ahead of their current location [63]. This is similar to look ahead timing seen in research on other motor actions, such as stair climbing, suggesting that this timing plays an important role in human movement [72].

Vision sensor is a user-independent method to explicitly sense the environment. It can be used to detect changes in terrain in advance relative to EMG activity or mechanical sensors measuring body kinematics (Fig 6.1).

With improved sensing and miniaturized computing in recent years, vision as a sensing modality has been increasingly researched for improving prosthetic control [62, 52, 114, 107]. A few practical challenges still remain before commercial adoption of vision in prosthesis.

A common approach is to train a classifier on images as inputs and corresponding mode labels as targets. Studies currently employ manual methods of labelling, which could be based on foot placement (kinematics) or based on visible features in image data. This entrenches subjectivity particularly near transitions[52]. Moreover, this process is time and resource

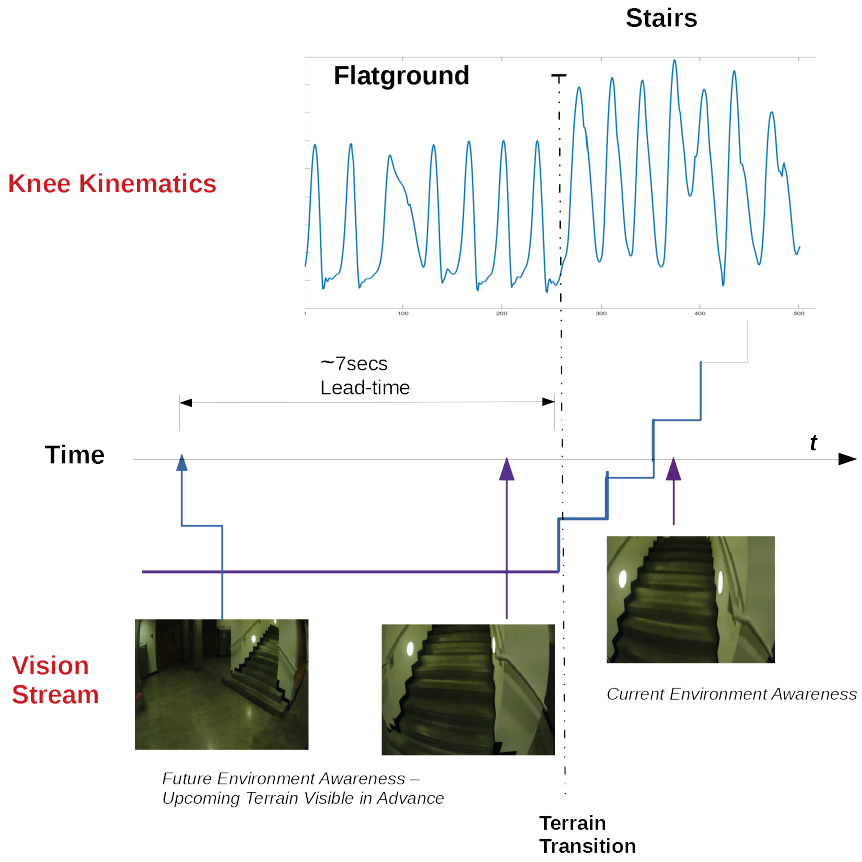


Figure 6.1: Vision sensor can foresee a new terrain before body kinematics change. In this example, the stair ascent was in the visual data stream almost 7 seconds prior to navigating the stairs. Vision can also provide information about current environment to provide fault tolerance and improve robustness

consuming, even for modest number of modes. For example, [52] 37000 vision frames were manually labeled for data from single subject with 3 mode classes.

Similar to EMG based approach, machine learning is employed for pattern recognition of sensor data for mode classification. Deep-learning models are the current state-of-the-art for image classification[33, 48]. Best results in mode-classification using vision have also leveraged variants of deep-learning models [117, 114]. Given the rapid advancement in embedded technology for deep-learning inference, it seems likely that future studies would choose this strategy as well. Table 6.1 summarizes the studies using vision as sensor for explicit environment sensing.

Practical application of deep-learning requires large training data-set. Comparing other domains which use vision for robotic control [56], the size of training data in the current prosthetic studies have been far fewer. This can limit generalization as the deep-learning models learn only the features pertaining to the limited training data.

6.1.2 Unsupervised Labeling and Transfer Learning for Prosthesis Control

In the previous chapter, we demonstrated a method for unsupervised labeling of kinematic training data from a clustering model. Here, we expand the same process to apply the model generated training labels for images. This allows training a vision classifier without the need for manual labeling.

We use *Transfer Learning*, a technique used in machine learning to accelerate learning and improve generalizability [93, 70]. Transfer learning paradigm leverages the knowledge acquired for one task to solve related ones. Generally, the source model is trained with a larger data-set and applied to a task with limited training examples. [82, 46].

The Convolutional Neural Network classifier used in this study is pre-trained on a million images from ImageNet dataset [85]. By transferring the learned features from this extensive dataset, the relatively small training set can still yield high classification accuracy. We compare the performance achieved with transfer learning with a model trained purely on our data.

We report lead time of labels predicted by vision classifier compared to kinematic based labels. The lead time is defined as the time elapse between when the vision system recognizes the change of environments and when the body kinematics reflect the change. We report overall and steady-state classification accuracy of the classifier for a test set with manually labeled ground truth. To summarize, in Aim 3a and 3b we

1. Demonstrate an unsupervised method to automatically label gait training data.

2. Apply the model-generated labels as targets to train a vision classifier.

3. Quantify the lead time using the vision classifier compared to mechanical sensor (kinematic) data.

4. Analyse the knee kinematics of each of the cluster groups and examine how the algorithm can help inform the appropriate number of modes present in the data.

6.1.3 Overview of Vision to Kinematics

Training: Translating vision directly to associated kinematic behaviour is hard. Instead, a CNN classifer can be trained to classify image data to locomotion mode labels (Fig 6.2a). The target labels for images are derived from clustering knee gait cycles (blue arrow) in the training set. Each cluster group comprises of gait cycles clustered based on similarity. Since kinematically similar behavior will be needed to locomote similar terrains, the cluster group labels are regarded as the locomotion mode labels.

Inference Unseen images are forwarded to the CNN classifier to predict vision labels $\hat{V}_t$ (Fig 6.2b). Although the *CNN classifier was trained on kinematic based labels, predictions are made based on visual similarities to the training images.* Since a new terrain will be visible before body kinematics adapt, the predicted vision label will lead the kinematic based $\hat{K}_t$. This key assumption allows anticipation of terrain transition before the user body reacts to the change.

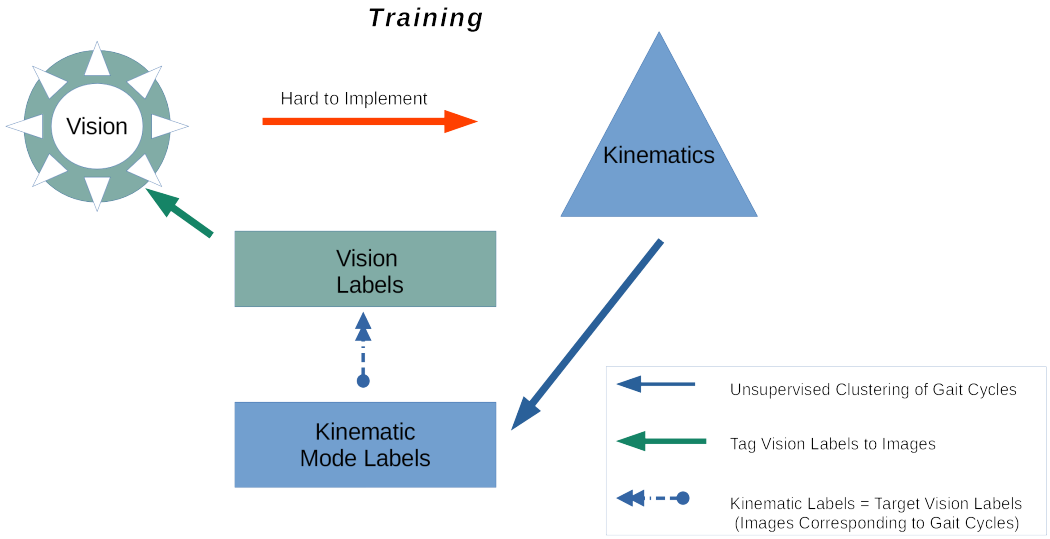
6.2 Methods

6.2.1 Data Collection and Experiments

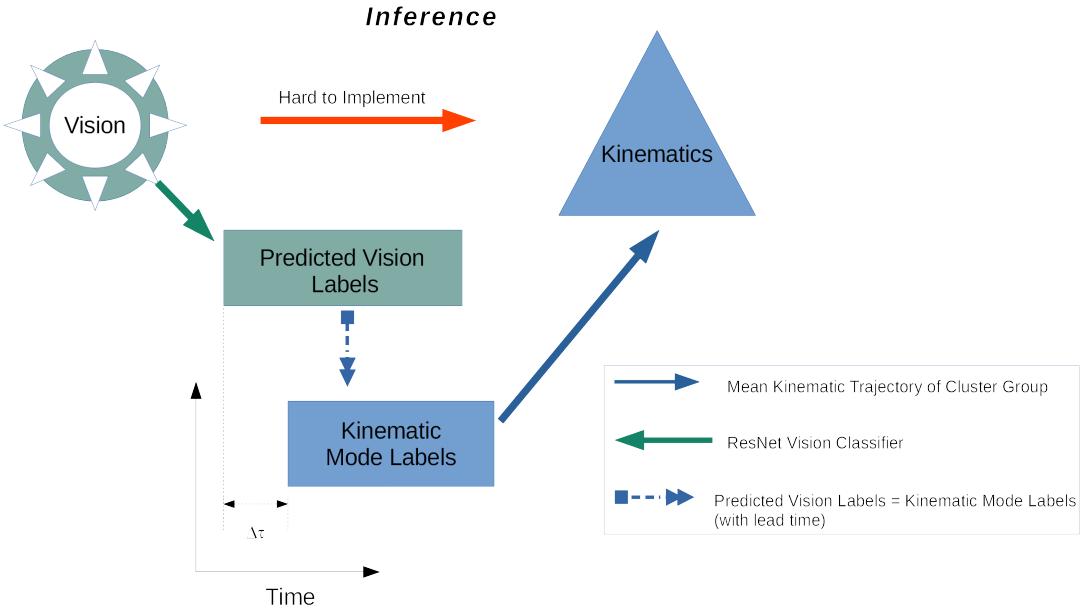
Ambulation data was collected for a total of 10 healthy participants with no amputation or other mobility impairments. Recruitment and human subject protocols were performed in accordance with the local University of Washington Institutional Review Board approval and each subject provided informed consent. De-identified data can be made available, via a data use agreement, upon request to the authors.

The subject's anthropometric details were recorded and 17 wearable sensors were placed on the their body. This was followed by a calibration procedure and brief test to see quality of data being recorded. The subjects then performed 10-15 minute trials of the desired activity. A head mounted ego-centric camera from Pupil Labs [45] was calibrated and used to collect visual data. The subjects were instructed to walk naturally at self-selected pace. The data for these activities was collected in public spaces during active business hours, with the intent that normal gait dynamics and corrections would appear in the example data.

Each trial started with participants performing flatground walking in a cluttered classroom environment with chairs, followed by sections of flatground walking in open corridors. The participants were then instructed to ascend a flight of stairs to reach the next level, which involved flatground walking in corridors. Participants returned to the original level by descending the flight of stairs. There were brief sections of atypical movements such as opening the classroom door to enter the corridor section, short sections (2-3 steps) of flatground in between flight of stairs.



(a) Process for training the vision classifier using labels from the clustering model.



(b) Process for inferring mode labels from the trained vision classifier. These predicted labels rely on visible features and will lead body movements

Figure 6.2: Translating vision directly to associated kinematic behaviour is hard. Instead, image data can be classified into locomotion mode classes, each with an associated kinematic profile. For training the vision classifier, the target labels are acquired from clustering knee

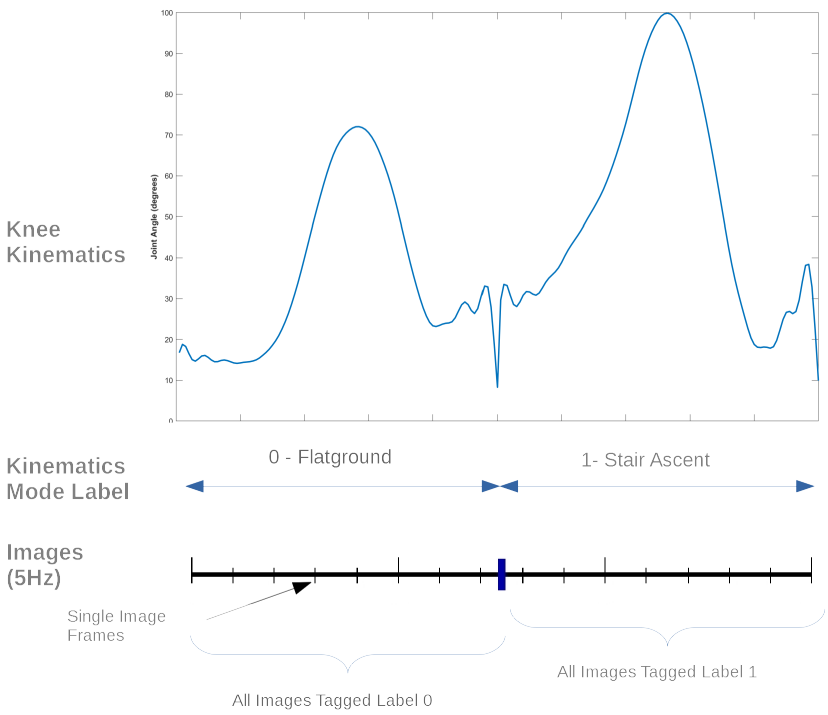


Figure 6.3: Each gait cycle has a label. All images corresponding to the duration of the gait cycle are tagged the same label

6.2.2 Instrumentation

We collected locomotion data using the Xsens Awinda suit [4], consisting of 17 body-worn sensors placed at key locations. Each sensor has a tri-axial gyroscope, accelerometer, magnetometer, and barometer. Xsens Analyse software integrates these individual sensors and renders a full-body avatar. After a system specified calibration, the software provides position and joint kinematics in a 3D environment. Although other data such as limb-segment position, orientation, acceleration are available, we used only joint angles for this study. All angles are in 1x3 Euler representation of the joint angle vector (x, y, z) in degrees, calculated using the Euler sequence ZXY using the International Society of Biomechanics standard joint angle coordinate system [105]. Data, sampled at 60 Hz, from a total of 22 joints in 3 anatomical planes (sagittal, frontal, transverse) were captured for each trial, which results in 66 total possible features for our machine learning methods.

6.2.3 Image Labeling

Image data was collected using the head mounted camera and down-sampled to 5Hz. Unsupervised cluster model was used to generate mode labels for every gait cycle (See Chapter

5). An average human gait cycle duration is about 1second[67]. At sample rate of 5Hz for image data, a single gait cycle contains about 5 images. All images corresponding to the duration of the gait cycle are tagged the same label as shown in Fig 6.3.

6.2.4 Machine Learning

Models

To improve generalization and to offset a relatively small training data size, we apply a technique called Transfer Learning. This is achieved by training a model on a large dataset (eg. ImageNet) and then transferring the learned knowledge (features, weights) by fine-tuning the model to a different dataset. This approach has shown to improve efficiency and performance as the network is not trained from scratch. More importantly, the features learned from the large dataset usually carries over to the new dataset and improves accuracy compared to a randomly initialized model[46].

The pre-trained model is adapted to the new dataset by training only the last layer, known as the head, responsible for generating the class labels. The rest of the model, known as the body or backbone, is frozen to retain the learning from the large dataset. If more precision is desired, the model can be fine-tuned by unfreezing the body and retraining the whole model but with low learning rate. This low learning rate ensures that model is not drastically perturbed while fine-tuning of the layers.

In this study we use a ResNet-18 a convolutional neural network that is 18 layers deep[33]. The network is pre-trained on more than a million images from the ImageNet database [85] with images from 1000 object categories, such as keyboard, mouse, pencil, and many animals. As a result, the network has learned rich feature representations for a wide range of images. We fine-tune this network to classify terrains into 3 classes(flatground, stair-ascent, and stair descent). The network has an image input size of 224-by-224. For comparision, we also report results from a randomly initialized ResNet-18 model.

Data

Kinematic and Vision Labels: Current research on vision for prosthesis distinguishes between 2 types of labels or ground truth for the data. Terrain labels can be based on

1. Current kinematic behaviour which generally uses foot position.

2. Visible features of the terrain in image data.

Since visible features of the new terrain can be in the image from a long distance, vision based labeling is prone to subjective bias[62, 52]. More importantly, for our purpose, this also means that vision based label will generally precede kinematics based labels. This lead-time is the main benefit of using vision to anticipate transitions

Steady-State and Transition Data: A common trend amongst in prosthetic control research is to separate steady-state and transition sections [42, 62]. Steady-state comprises sections of data without any terrain or activity changes, where a periodic gait cycle is repeated. Transitions consist of sections with locomotion mode changes. The performance evaluation of transition, however, is less clearly established, which we discuss below.

Training and Inference Process

Training: The training process uses labels generated by the unsupervised model to train the vision classifier.

1. Knee gait cycles in the training dataset are clustered into optimal number of clusters using an unsupervised clustering algorithm (training-blue arrow). Each group is assigned a kinematic mode label.

2. Each gait cycle has several corresponding images (Fig 6.3). The label associated with a gait cycle is used to tag all the corresponding images.

3. A CNN classifier is trained on the images as inputs and tagged labels as targets(training-green arrow)

Inference: Since a new terrain will be visible before body kinematics adapt, the predicted vision label will lead the kinematic based $\hat{K}_t$. We define the lead time as

$$\Delta t = \mid \hat{V}_t - \hat{K}_t \mid$$

1. During inference, images are forwarded to the CNN classifier to predict vision labels$\hat{V}_t$ (inference-green arrow).

2. For steady-state gait, the predicted $\hat{V}_t$ will match the $\hat{K}_t$ mode label.

3. For an upcoming transition, new terrain will visible prior to the EMG or kinematics change to adapt. Hence the mode labels predicted by CNN classifier precede kinematic based labels. That is $\hat{V}_t$ will match the mode class of $\widehat{K_{t+\Delta t}}$

4. Each mode label was originally derived from a cluster group comprising of several samples of gait cycles clustered based on similarity. The spatial average of the samples of a group can be used as representative reference trajectory for that mode label (e.g. flatground). In our study this reference pertains to knee joint angle trajectory, however any behaviour can be associated to a mode label.

Training with Steady-State Data Only: The approach described above relies on transference of kinematic based labels to vision data. Kinematic sensor data are inherently delayed and hence the labels represent the mode class after transitioning to new terrain type. As such, this labeling scheme is only valid for steady-state sections. Hence, we train the vision classifier on steady-state data only. About 3 seconds of data prior to transitions are considered 'transition' data and omitted from training. This allows the classifier to predict transition sections without a firm correlation to kinematic labels. This behaviour is desired for a larger lead-time. Consequently, while evaluating performance and reporting steady-state accuracy, transition sections are omitted.

6.2.5 Performance Evaluation and Analysis

Vision based prosthesis studies employ 3 evaluation metrics to evaluate various characteristics of resulting performance. Overall classification accuracy (CA) is the percentage of accurately labeled images relative to all images in the test data. Steady- state classification accuracy is similar but omits transitions sections of the data.

Evaluation of performance for transitions is not clearly defined and several methods are observed in studies (Table 6.1). In EMG studies [111, 110], specific gait events such as toe off from the level ground to upstairs and heel contact from down stairs to the level ground are used to estimate the deadline of locomotion transition. Prediction lead time with respect to this deadline is used to quantify the anticipatory response of the system. [107] use a similar approach by segmenting gait and use mid swing as the critical deadline. However, they report the percentage of transitions detected before the foot lift of the leading limb. [114] report lead time for a CNN based vision classifier with respect to kinematic based labels. [117] defined the transition period as the 5 second period before a transition and report terrain prediction accuracy during these periods. [62] report percentage of total transitions detected.

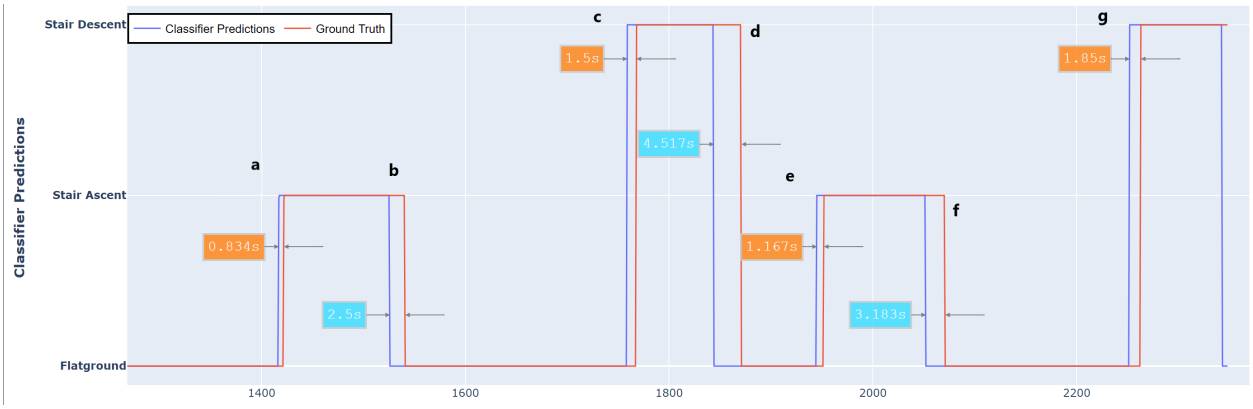
We report overall CA, steady-state CA and the lead time of the vision based label with respect to the kinematic labels for every terrain type.

6.3 Results

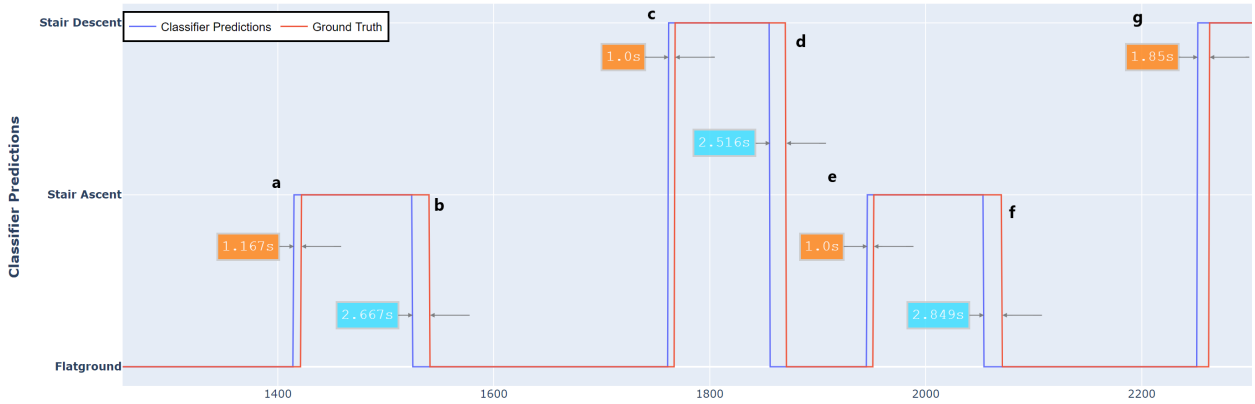
Performance of the vision classifier was evaluated on test set with manually labeled ground truth. Fig 6.4 shows the predicted and actual mode labels for unseen test subject data. Transitions and lead-times are also shown.

6.3.1 Classification Accuracy

The classifier pre-trained on the ImageNet data (containing a million images) achieved an overall CA of 96.6% and steady-state CA of 99.5%. In comparison, the classifier initialized with random weights had an overall CA of 95.7% and a steady-state accuracy of 99.14%.



(a) Base Model:Predicted mode labels for an unseen test set using a randomly initialized ResNet-18 model.



(b) Transfer-Learning Model: Predicted mode labels for an unseen test set using a ResNet-18 model pre-trained on ImageNet data.

Figure 6.4: Predicted and actual mode labels for an unseen test subject data. Transfer learning improved overall classification accuracy and average transition lead-time. Lead-time is defined as the time elapsed between the vision system detecting a change in terrain and the body kinematics of the user changing to adapt to the new terrain.

	Base Model	Tranfer-Learning Model
a	0.83s	1.17s
b	2.5s	2.67s
c	1.5s	1.0s
d	4.517s	2.52s
e	1.167s	1.0s
f	3.183s	2.849s
g	1.85s	1.85s

Table 6.2: Transition lead-times for the base model (with randomly initialized weights) and the transfer-learning model (with pre-trained weights). Base model had a lower average lead time of 1.08 seconds compared to 1.8seconds achieved by the transfer-learning model. Certain particular transitions such as stair-ascent to flat ground and stair-descent to flatground had relatively better lead time with the base model.

We see that transfer learning does improve the performance but the test data size is limited to make an accurate assessment. We discuss this in the next section

The system noted 99% steady-state and 96% overall accuracy of mode classification. Transitions were detected an average of 1.8 secs before kinematics changed to adapt to the new terrain.

6.3.2 Transitions

The classifier pre-trained on the ImageNet data had an average lead time of 1.8 seconds, while the base model (randomly initialized) had an average lead time of 1.08 seconds. Once again, tranfer learning approach improved performance. Interestingly, however, lead-times for some of the transitions such as transitioning to flat-ground from stair ascent (f) and stair descent (d) was significantly better for the base model (Table 6.2).

6.4 Discussion

The aims described in this chapter and the previous chapter 5 together demonstrate a method for training a vision classifier without the intensive process of manually labeling the responses. We show that explicit environment sensing would enable detection of transitions almost a full step in advance. This way of anticipating terrain changes would allow the prosthesis to execute a smooth transition to the desired locomotion mode.

Vision being a user-independent sensing modality also allows better error-correction by not relying solely on user actions for the same.

The system noted a high steady-state classification of 99% for 3 modes. But as authors of [52] noticed, individual mode accuracy matters. Most datasets, including ours, are unbalanced consisting mostly (80%+) of flatground.

We aimed to improve generalizability by leveraging knowledge learned from other datasets. The transferred model was pre-trained on a million images of the ImageNet dataset. This method has been known to improve performance when the target dataset is limited. This transferred model showed a slight improvement over the randomly generated base model. However, the test data used to evaluate our models was collected in the same environment as the training set. A test set with different environmental features would be a true test of the generalizability of transfer learning.

The average transition performance of the transferred model was better than the randomly initialized base model (1.8secs vs 1.08). Interestingly, however, certain transitions such as stair descent to flat ground showed larger lead times with the base model. We expect this to be a result of the model over-fitting to the particular scenario. As discussed above, the test set and training set are quite similar in their visual features. A more diverse environment as the test data will shed more light on this performance boost seen in the case of the base model.

6.5 Conclusion

Humans rely on vision to navigate and adapt to the complex and ever-changing environments surrounding us. Current prosthesis control could benefit from the same environment awareness. However, training vision classifiers require a significant amount of training data and resources to manually label the desired responses. As our primary aim 3, we address both these challenges. We present a novel method to acquire labels without much human supervision in the previous chapter. We apply these labels to train a vision classifier that leverages knowledge learned from a large public dataset. We show that terrains can be accurately estimated 99% of the time and transitions can be anticipated almost 1.8 seconds in advance. This could enable prosthesis to implement smooth and safe control.

Chapter 7

REAL-TIME TESTS OF THE COORDINATED MOVEMENT CONTROLLER

In this chapter we extend the analysis and results described in the previous chapter, towards actively controlling a powered limb in real time.

7.1 Experiment Setup

The pipeline of the Coordinated Movement (CM) controller entails 1) acquisition of kinematics from Xsens, 2) pre-processing of the data (See Section 4.2.6 a,b), 3) predicting the prosthetic joint trajectories by the neural network, and finally 4) translating these predictions for lower-level control of the Open Source Leg (OSL). The lack of HD processing (See Section 4.2.5) of kinematics can introduce noise and error in predictions. The pre-processing of data and predictions are operations that take finite time. A delay or error in response of a load-bearing prosthesis could result in critical injuries in a real-life scenario. This real-time test demonstrates the overall latency in the CM pipeline as well the quality of live predictions and evaluates the safety of the CM controller prior to actual subject experiments.

We used the Open Source Leg (OSL) [10], a modular lightweight robotic leg designed to facilitate a common hardware test-bed for prosthetic control research. It provides an open source API to control position, torque and impedance of the knee and ankle joint making it an ideal platform to compare different control strategies.

A person with no amputation wearing an Xsens motion capture suit ambulated on a treadmill at a self-selected speed (Fig 7.1). A real-time controller built using Xsens Python SDK acquired live kinematics as inputs. Right ankle and knee joints were omitted as they were the intended target prosthetic joints. An LSTM neural network (see Section 4.2.6) pretrained on offline flat-ground walking data (to match the treadmill walking) was used to predict right ankle and knee joint kinematics from the inputs. These predicted joint kinematics were encoded to actuator positions on the Open Source Leg. The controller ran on a GPU powered laptop tethered to the OSL.

7.2 Results

A single individual who was not a part of the training cohort was used for a systems test. Kinematics were collected as they walked on a treadmill at self selected pace. These data were played as inputs to predict ankle and knee joint trajectories for the Coordinated Movement controller for a prosthetic limb in real-time (Fig. 7.2). For the knee joint, the average

Figure 7.1: Real-time setup with Open Source Leg (OSL). An individual wearing the motion capture sensors walked on the treadmill at self-selected speed. Live kinematics from the suit were used as inputs to a pre-trained network that generated right ankle and knee predictions. These predictions were used to actuate the OSL in real time.

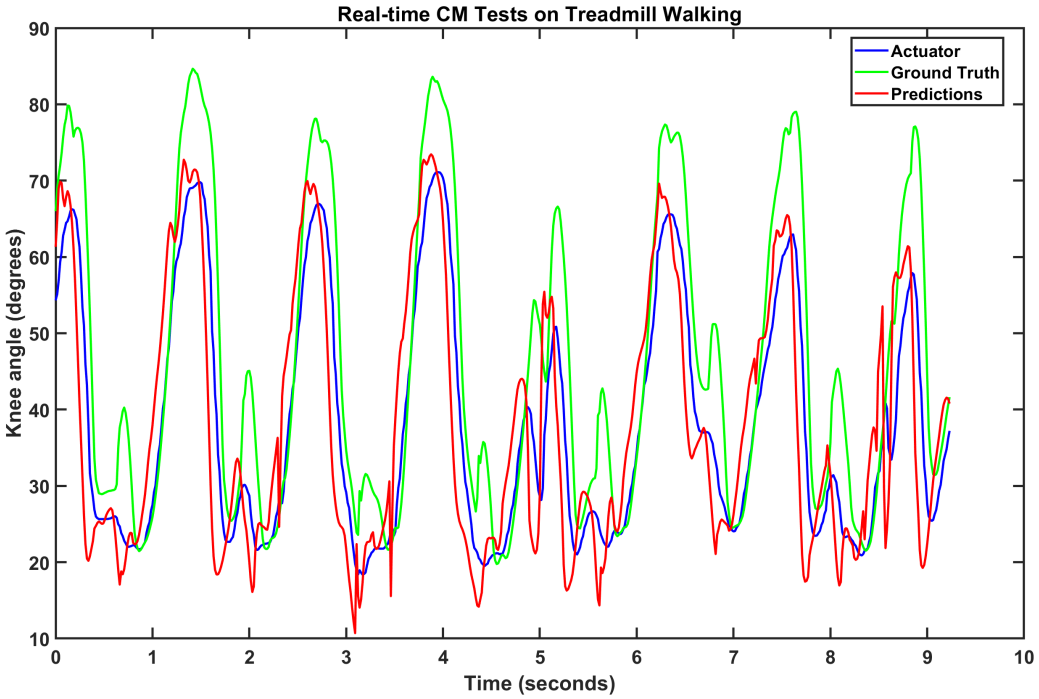


Figure 7.2: Knee joint predicted (red) and actuated trajectories (blue) during the real-time tests with treadmill walking activity. For this trial, predicted trajectory had an RMS error of 17.62 degrees and the actuated trajectory had an error of 10.38 degrees, with respect to the actual trajectory executed by the subject(green).

RMS error was 17.62 degrees for the network predictions and 10.38 degrees for the actuated trajectory. These errors were higher than the offline results seen in Fig. 4.4 for flat ground activity. The error was calculated with respect to the actual knee trajectory measured on the test subject, which represents the ideal or the desired trajectory.

7.3 Discussion

The real-time test demonstrated the feasibility of our controller to actuate a prosthetic leg. The RMS error of the real-time predictions were slightly greater than those observed in offline analysis, underscoring the role of algorithmic data-processing in wearable motion capture. Raw motion-tracking data is inherently noisy. Xsens real-time engine mitigates this by accommodating sensor drifts and correlating independent sensor data to a human body model. A post processing engine further improves data quality by including past, present, and future samples. While offline analysis has the benefit of using the clean post processed

data, the real-time control does not. This results in poorer prediction performance. This deviation in distribution of the data in test compared to training, known as covariate shift, is commonly seen in machine learning. This could be alleviated by using raw unprocessed data, or processed data with artificial noise, as training inputs to simulate the real-time data. Generation of synthetic training data using Generative Adversarial Networks (GAN) could allow the network to be more robust to sensor noise[39]. A more thorough long term solution could be to engineer an algorithmically light version of the post processing engine to operate on real-time data as well. The whole pipeline of operations resulted in a lag of less than 0.05 seconds in the response of the prosthetic leg. For upper-limb protheses a delay greater than 300ms is considered significant [18]. However, this value has not been established for lower-limb prostheses.

An impedance or torque based prosthetic controller is likely to be more comfortable relative to position control. Current commercial wearable sensors provide only position estimates, limiting the target predictions in this study to joint kinematics only. However, to replicate force plate data in outside laboratory conditions there is an increased interest in estimating joint kinetics using wearable sensors[7, 59]. Upon availability of such reference data, this data driven methodology can be used to generate joint torque predictions as well.

7.4 Conclusion

Real-time tests provided valuable insights into user experience for the CM controller. The value of clean input kinematics was underscored. An impedance or torque based control may provide a smoother user experience and should be explored as an alternative to the position control strategy demonstrated here.

Chapter 8

CONCLUSION

Today's prostheses artificially constrain movements of lower limb prostheses, forcing users to shape their life around what their leg CAN'T do. In my work, I've investigated how to break the mold using several state-of-the-art techniques. First, I've demonstrated that vision-based sensing and control can make prostheses anticipate the terrain around them with high accuracy, even in terrains they haven't seen before (Fig.6.4). Second, I've shown the potential of using the body's own intact movements as a guide for continuously controlling the prosthesis - opening up the full range of human movements (Fig 4.5). With these results, we can envision a path forward where users of lower limb prostheses instead shape their life around what their leg CAN do.

To improve sensing, we obtain environment information while reducing the over reliance on user-dependent sensors. User dependent sensors focus more on user state, even for environment information. This can increase the cognitive burden on the user. Safety is also compromised as the controller has to rely on user movements to recover from errors. The error causes disturbance in the user state and reliance on this disturbed state to recover can result in a vicious cycle of errors. User dependent sensors also need user-centric calibration and customization which require a trained prosthetist.

Vision as a user independent sensing modality addresses a lot of these limitations. We show 2 ways of using vision for environment sensing. Both methods explicitly sense the environment and can complement user dependent sensors. Such a complementary system will suffer from different sources of errors, improving the overall system robustness. Individual user based customization is also minimized which could presumably allow off-the-shelf deployment.

Towards a new control paradigm we demonstrate a continuous controller which is a radical shift from the current discrete mode-based strategy. Without categorization of movements, this coordinated movement controller provides continuous commands without explicit transition mechanisms. Change from one type of movement to another occurs naturally, using the body inter-joint coordination.

A key benefit of the data-driven strategy is the scalability to add new movements on demand, without compromising the performance of existing movements. A neural network can be retrained with data comprising current and new movements. This can be done offline and downloaded on to the device to incorporate a richer repertoire of movements.

Although our aims focus on sensors and control strategies separately, they both form an integral part of the user experience. Future work should focus on a framework integrating vision with mode-based or continuous control strategies.

With over a million amputees across the globe living in different socio-economic and cultural backgrounds, a standard one-size-fits-all solution is a futile quest. In regions with low availability of individualized health-care personnel (e.g, trained prosthetist), user-independent and off-the-shelf sensing may be desirable. Safe and predicable behaviour of mode based control could be desirable by high fall risk or elderly users with low mobility. Low mobility users with few frequently visited areas can also benefit more from localization based application of vision than the more generalized terrain recognition one.

Affordability will be a major factor for most of the world population. A low-functioning cheap prosthesis with a handful of mode based movements may suit the needs of a few. A high functioning continuous control with an option of downloading movements to suit the environment and profession may be the choice of others.

In this dissertation, we provide different avenues towards a more life-like prosthesis for powered limb users across the globe. Although many hurdles still remain, we hope that pieces of this research will empower someone to walk, or perhaps even dance, again.

BIBLIOGRAPHY

[1] Here positioning — global positioning solutions for business. https://here.com/en/products-services/products/here-positioning. (Accessed on 04/24/2017).

[2] Indoor maps – about – google maps. https://www.google.com/maps/about/partners/indoormaps/. (Accessed on 04/25/2017).

[3] Microprocessor-controlled lower limb prostheses. https://www.hca.wa.gov/assets/program/mclowerprostheticfinalreport[1].pdf. (Accessed on 03/30/2020).

[4] Mtw awinda - products - xsens 3d motion tracking. https://www.xsens.com/products/mtw-awinda/. (Accessed on 10/29/2018).

[5] The next frontier of navigation: In-location positioning — wired. https://www.wired.com/insights/2013/06/the-next-frontier-of-navigation-in-location-positioning/. (Accessed on 04/24/2017).

[6] What is wifislam and why did apple want it? https://thenextweb.com/apple/2013/03/26/what-exactly-wifislam-is-and-why-apple-acquired-it. (Accessed on 04/24/2017).

[7] Andrea Ancillao, Salvatore Tedesco, John Barton, and Brendan O'Flynn. Indirect measurement of ground reaction forces and moments by means of wearable inertial sensors: A systematic review. *Sensors*, 18(8):2564, 2018.

[8] P Artemiadis. Emg-based robot control interfaces: past, present and future. *Advances in Robotics & Automation*, 1(2):1–3, 2012.

[9] Samuel Au, Max Berniker, and Hugh Herr. Powered ankle-foot prosthesis to assist level-ground and stair-descent gaits. *Neural Networks*, 21(4):654–666, 2008.

[10] Alejandro F Azocar, Luke M Mooney, Levi J Hargrove, and Elliott J Rouse. Design and characterization of an open-source robotic leg prosthesis. In *2018 7th IEEE International Conference on Biomedical Robotics and Biomechatronics (Biorob)*, pages 111–118. IEEE, 2018.

[11] Alexander V Bates, Alison H McGregor, and Caroline M Alexander. Reliability and minimal detectable change of gait kinematics in people who are hypermobile. *Gait & posture*, 44:37–42, 2016.

[12] N Alberto Borghese, L Bianchi, and F Lacquaniti. Kinematic determinants of human locomotion. *The Journal of physiology*, 494(3):863–879, 1996.

[13] Starr E Brown, Jason M Wilken, Elizabeth Russell Esposito, Stefania Fatone, and Andrea J Ikeda. Evaluation of nu-flexsiv socket performance for military service members with transfemoral amputation. *US Army Medical Department journal*, 2018.

[14] Dustin A Bruening and Adam M Fullenkamp. Sex differences in whole body gait kinematics at preferred speeds. *Gait and posture*, 41(2):540–545, 2015.

[15] J-JJ Chen and Richard Shiavi. Temporal feature extraction and clustering analysis of electromyographic linear envelopes in gait studies. *IEEE Transactions on Biomedical Engineering*, 37(3):295–302, 1990.

[16] Jacob Cohen. Statistical power analysis. *Current directions in psychological science*, 1(3):98–101, 1992.

[17] MWMG Dissanayake, Paul Newman, Steven Clark, Hugh F Durrant-Whyte, and Michael Csorba. A solution to the simultaneous localization and map building (SLAM) problem. *Robotics and Automation, IEEE Transactions on*, 17(3):229–241, 2001.

[18] Kevin Englehart and Bernard Hudgins. A robust, real-time control scheme for multifunction myoelectric control. *Biomedical Engineering, IEEE Transactions on*, 50(7):848–854, 2003.

[19] Dario Farina, Roberto Merletti, Barbara Indino, Marisa Nazzaro, and Marco Pozzo. Surface EMG crosstalk between knee extensor muscles: experimental and model results. *Muscle & nerve*, 26(5):681–695, 2002.

[20] Madalina Fiterau, Suvrat Bhooshan, Jason Fries, Charles Bournhonesque, Jennifer Hicks, Eni Halilaj, Christopher Ré, and Scott Delp. Shortfuse: Biomedical time series representations in the presence of structured information. *arXiv preprint arXiv:1705.04790*, 2017.

[21] Udo Frese. Interview: Is SLAM Solved? *KI - Künstliche Intelligenz*, 24(3):255–257, jul 2010.

[22] L. Gabert, S. Hood, M. Tran, M. Cempini, and T. Lenzi. A compact, lightweight robotic ankle-foot prosthesis: Featuring a powered polycentric design. *IEEE Robotics Automation Magazine*, 27(1):87–102, 2020.

[23] Robert S Gailey, Charles Scoville, Ignacio A Gaunaurd, Michele A Raya, Alison A Linberg, Paul D Stoneman, Stuart M Campbell, and Kathryn E Roach. Construct validity of comprehensive high-level activity mobility predictor (champ) for male servicemembers with traumatic lower-limb loss. *Journal of Rehabilitation Research & Development*, 50(7):919–931, 2013.

[24] H. Gao, L. Luo, M. Pi, Z. Li, Q. Li, K. Zhao, and J. Huang. Eeg-based volitional control of prosthetic legs for walking in different terrains. *IEEE Transactions on Automation Science and Engineering*, 2019.

[25] Michael Goldfarb, Huseyin Atakan Varol, Frank Charles Sup IV, Jason Mitchell, and Thomas J Withrow. Powered leg prosthesis and control methodologies for obtaining near normal gait. 2015.

[26] Google. ATAP Project Tango – Google, February 2014.

[27] Donald Lee Grimes. *An active multi-mode above knee prosthesis controller*. PhD thesis, Massachusetts Institute of Technology, 1979.

[28] Brian J Hafner, Sara J Morgan, Daniel C Abrahamson, and Dagmar Amtmann. Characterizing mobility from the prosthetic limb user's perspective: Use of focus groups to guide development of the prosthetic limb users survey of mobility. *Prosthetics and orthotics international*, 40(5):582–590, 2016.

[29] Alon Halevy, Peter Norvig, and Fernando Pereira. The unreasonable effectiveness of data. *IEEE Intelligent Systems*, 24(2):8–12, 2009.

[30] Maria Halkidi, Yannis Batistakis, and Michalis Vazirgiannis. Cluster validity methods: part i. *ACM Sigmod Record*, 31(2):40–45, 2002.

[31] Levi J Hargrove, Ann M Simon, Aaron J Young, Robert D Lipschutz, Suzanne B Finucane, Douglas G Smith, and Todd A Kuiken. Robotic leg control with EMG decoding in an amputee with nerve transfers. *New England Journal of Medicine*, 369(13):1237–1242, 2013.

[32] Levi J Hargrove, Aaron J Young, Ann M Simon, Nicholas P Fey, Robert D Lipschutz, Suzanne B Finucane, Elizabeth G Halsne, Kimberly A Ingraham, and Todd A Kuiken. Intuitive control of a powered prosthetic leg during ambulation: a randomized clinical trial. *JAMA*, 313(22):2244–2252, 2015.

[33] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[34] M Jason Highsmith, Jason T Kahle, Stephanie L Carey, Derek J Lura, Rajiv V Dubey, and William S Quillen. Kinetic differences using a power knee and c-leg while sitting down and standing up: a case report. *JPO: Journal of Prosthetics and Orthotics*, 22(4):237–243, 2010.

[35] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[36] Sungchul Hong, Jaehoon Jung, Sangmin Kim, Hyoungsig Cho, Jeongho Lee, and Joon Heo. Semi-automated approach to indoor mapping for 3D as-built building information modeling. *Computers, Environment and Urban Systems*, 51:34–46, 2015.

[37] C. D. Hoover, G. D. Fulk, and K. B. Fite. Stair ascent with a powered transfemoral prosthesis under direct myoelectric control. *IEEE/ASME Transactions on Mechatronics*, 18(3):1191–1200, 2013.

[38] Carl D Hoover, George D Fulk, and Kevin B Fite. The design and initial experimental validation of an active myoelectric transfemoral prosthesis. *Journal of Medical Devices*, 6(1):011005, 2012.

[39] B. Hu, A. M. Simon, and L. Hargrove. Deep generative models with data augmentation to learn robust representations of movement intention for powered leg prostheses. *IEEE Transactions on Medical Robotics and Bionics*, 1(4):267–278, Nov 2019.

[40] He Huang, Zhi Dou, Fan Zhang, and Michael J Nunnery. Improving the performance of a neural-machine interface for artificial legs using prior knowledge of walking environment. In *Engineering in Medicine and Biology Society, EMBC, 2011 Annual International Conference of the IEEE*, pages 4255–4258. IEEE, 2011.

[41] He Huang, Todd A Kuiken, and Robert D Lipschutz. A strategy for identifying locomotion modes using surface electromyography. *Biomedical Engineering, IEEE Transactions on*, 56(1):65–73, 2009.

[42] He Huang, Fan Zhang, Levi J Hargrove, Zhi Dou, Daniel R Rogers, and Kevin B Englehart. Continuous locomotion-mode identification for prosthetic legs based on neuromuscular–mechanical fusion. *Biomedical Engineering, IEEE Transactions on*, 58(10):2867–2875, 2011.

[43] Wenchao Jiang and Zhaozheng Yin. Human activity recognition using wearable sensors by deep convolutional neural networks. In *Proceedings of the 23rd ACM international conference on Multimedia*, pages 1307–1310. Acm, 2015.

[44] Deepak Joshi and Michael E Hahn. Terrain and Direction Classification of Locomotion Transitions Using Neuromuscular and Mechanical Input. *Annals of biomedical engineering*, 44(4):1275–1284, 2016.

[45] Moritz Kassner, William Patera, and Andreas Bulling. Pupil: An open source platform for pervasive eye tracking and mobile gaze-based interaction. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication*, UbiComp '14 Adjunct, pages 1151–1160, New York, NY, USA, 2014. ACM.

[46] Simon Kornblith, Jonathon Shlens, and Quoc V Le. Do better imagenet models transfer better? In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2661–2671, 2019.

[47] Nili Eliana Krausz, Tommaso Lenzi, and Levi J Hargrove. Depth Sensing for Improved Control of Lower Limb Prostheses. *Biomedical Engineering, IEEE Transactions on*, 62(11):2576–2587, 2015.

[48] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[49] J Kulkarni, WJ Gaine, JG Buckley, JJ Rankine, and J Adams. Chronic low back pain in traumatic lower limb amputees. *Clinical rehabilitation*, 19(1):81–86, 2005.

[50] Jai Kulkarni, Judith Adams, Elaine Thomas, and Alan Silman. Association between amputation, arthritis and osteopenia in british male war veterans with major lower limb amputations. *Clinical Rehabilitation*, 12(4):348–353, 1998.

[51] Daniël Lakens. Equivalence tests: a practical primer for t tests, correlations, and meta-analyses. *Social psychological and personality science*, 8(4):355–362, 2017.

[52] Brock Laschowski, William McNally, Alexander Wong, and John McPhee. Preliminary design of an environment recognition system for controlling robotic lower-limb prostheses and exoskeletons. In *2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR)*, pages 868–873. IEEE, 2019.

[53] Brian E Lawson, Huseyin Atakan Varol, Amanda Huff, Erdem Erdemir, and Michael Goldfarb. Control of stair ascent and descent with a powered transfemoral prosthesis. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 21(3):466–473, 2013.

[54] Mikhail A Lebedev, Andrew J Tate, Timothy L Hanson, Zheng Li, Joseph E O'Doherty, Jesse A Winans, Peter J Ifft, Katie Z Zhuang, Nathan A Fitzsimmons, David A Schwarz, et al. Future developments in brain-machine interface research. *Clinics*, 66:25–32, 2011.

[55] T. Lenzi, M. Cempini, L. J. Hargrove, and T. A. Kuiken. Design, development, and validation of a lightweight nonbackdrivable robotic ankle prosthesis. *IEEE/ASME Transactions on Mechatronics*, 24(2):471–482, April 2019.

[56] Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. *The Journal of Machine Learning Research*, 17(1):1334–1373, 2016.

[57] Guanglin Li, Yaonan Li, Long Yu, and Yanjuan Geng. Conditioning and Sampling Issues of EMG Signals in Motion Recognition of Multifunctional Myoelectric Prostheses. *Annals of Biomedical Engineering*, 39(6):1779–1787, 2011.

[58] Ming Liu, Ding Wang, and He Helen Huang. Development of an environment-aware locomotion mode recognition system for powered lower limb prostheses. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 24(4):434–443, 2016.

[59] Tao Liu, Yoshio Inoue, and Kyoko Shibata. A wearable ground reaction force sensor system and its application to the measurement of extrinsic gait variability. *Sensors*, 10(11):10240–10255, 2010.

[60] Simon Lynen, Michael Bosse, Paul Furgale, and Roland Siegwart. Placeless place-recognition. In *2014 2nd International Conference on 3D Vision*, volume 1, pages 303–310. IEEE, 2014.

[61] Ernesto C Martinez-Villalpando and Hugh Herr. Agonist-antagonist active knee prosthesis: a preliminary study in level-ground walking. *Journal of Rehabilitation Research & Development*, 46(3), 2009.

[62] Yerzhan Massalin, Madina Abdrakhmanova, and Huseyin Atakan Varol. User-independent intent recognition for lower limb prostheses using depth sensing. *IEEE Transactions on Biomedical Engineering*, 65(8):1759–1770, 2017.

[63] Jonathan Samir Matthis, Jacob L Yates, and Mary M Hayhoe. Gaze and the control of foot placement when walking in natural terrain. *Current Biology*, 28(8):1224–1233, 2018.

[64] José del R Millán, Rüdiger Rupp, Gernot Mueller-Putz, Roderick Murray-Smith, Claudio Giugliemma, Michael Tangermann, Carmen Vidaurre, Febo Cincotti, Andrea Kubler, Robert Leeb, et al. Combining brain–computer interfaces and assistive technologies: state-of-the-art and challenges. *Frontiers in neuroscience*, 4:161, 2010.

[65] Piotr Mirowski, Tin Kam Ho, Saehoon Yi, and Michael MacDonald. Signalslam: Simultaneous localization and mapping with mixed wifi, bluetooth, lte and magnetic signals. In *Indoor Positioning and Indoor Navigation (IPIN), 2013 International Conference on*, pages 1–10. IEEE, 2013.

[66] BC Muir, JM Haddad, Michel JH Heijnen, and S Rietdyk. Proactive gait strategies to mitigate risk of obstacle contact are more prevalent with advancing age. *Gait & posture*, 41(1):233–239, 2015.

[67] M Pat Murray. Gait as a total pattern of movement: Including a bibliography on gait. *American Journal of Physical Medicine & Rehabilitation*, 46(1):290–333, 1967.

[68] Francisco Javier Ordóñez and Daniel Roggen. Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition. *Sensors*, 16(1), 2016.

[69] Michael S Orendurff, Jason A Schoen, Greta C Bernatz, Ava D Segal, and Glenn K Klute. How humans walk: bout duration, steps per bout, and rest duration. *Journal of Rehabilitation Research & Development*, 45(7), 2008.

[70] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2009.

[71] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. In *NIPS-W*, 2017.

[72] Aftab E Patla and Joan N Vickers. How far ahead do we look when required to step on specific locations in the travel path during locomotion? *Experimental brain research*, 148(1):133–138, 2003.

[73] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[74] Louis Peeraer, B Aeyels, and Georges Van der Perre. Development of EMG-based mode and intent recognition algorithms for a computer-controlled above-knee prosthesis. *Journal of biomedical engineering*, 12(3):178–182, 1990.

[75] S.J. PENSACK-RINEHART, G. REANEY, Y. Chawathe, N. Lee, S.B. BRAWER, and P. MESSMER. Providing indoor map data to a client computing device, April 30 2015. WO Patent App. PCT/US2014/045,558.

[76] Bogdan Pogorelc and Matjaž Gams. Detecting gait-related health problems of the elderly using multidimensional dynamic time warping approach with semantic attributes. *Multimedia tools and applications*, 66(1):95–114, 2013.

[77] Sasanka Potluri, Arvind Beerjapalli Chandran, Christian Diedrich, and Lutz Schega. Machine learning based human gait segmentation with wearable sensor platform. In *2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 588–594. IEEE, 2019.

[78] D. Quintero, D. J. Villarreal, D. J. Lambert, S. Kapp, and R. D. Gregg. Continuous-phase control of a powered knee–ankle prosthesis: Amputee experiments across speeds and inclines. *IEEE Transactions on Robotics*, 34(3):686–701, 2018.

[79] V. Rai, A. Sharma, and E. Rombokas. Mode-free control of prosthetic lower limbs. In *2019 International Symposium on Medical Robotics (ISMR)*, pages 1–7, April 2019.

[80] Vijeth Rai and Eric Rombokas. A framework for mode-free prosthetic control for unstructured terrains. In *2019 IEEE 16th International Conference on Rehabilitation Robotics (ICORR)*, pages 796–802. IEEE, 2019.

[81] Michele A Raya, Robert S Gailey, Ignacio A Gaunaurd, Daniel M Jayne, Stuart M Campbell, Erica Gagne, Patrick G Manrique, Daniel G Muller, and Christen Tucker. Comparison of three agility tests with male servicemembers: Edgren side step test, t-test, and illinois agility test. *Journal of Rehabilitation Research & Development*, 50(7):951–961, 2013.

[82] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? *arXiv preprint arXiv:1902.10811*, 2019.

[83] Nils Reimers and Iryna Gurevych. Optimal hyperparameters for deep lstm-networks for sequence labeling tasks. *arXiv preprint arXiv:1707.06799*, 2017.

[84] S. Rezazadeh, D. Quintero, N. Divekar, E. Reznick, L. Gray, and R. D. Gregg. A phase variable approach for improved rhythmic and non-rhythmic control of a powered knee-ankle prosthesis. *IEEE Access*, 7:109840–109855, 2019.

[85] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

[86] Ann M Simon, Nicholas P Fey, Kimberly A Ingraham, Aaron J Young, and Levi J Hargrove. Powered prosthesis control during walking, sitting, standing, and non-weight bearing activities using neural and mechanical inputs. In *Neural Engineering (NER), 2013 6th International IEEE/EMBS Conference On*, pages 1174–1177. IEEE, 2013.

[87] Ann M Simon, Kimberly A Ingraham, Nicholas P Fey, Suzanne B Finucane, Robert D Lipschutz, Aaron J Young, and Levi J Hargrove. Configuring a powered knee and ankle prosthesis for transfemoral amputees within five specific ambulation modes. *PloS one*, 9(6):e99387, 2014.

[88] SS Srinivasan, MJ Carty, PW Calvaresi, TR Clites, BE Maimon, CR Taylor, AN Zorzos, and H Herr. On prosthetic control: A regenerative agonist-antagonist myoneural interface. *Science Robotics*, 2(6), 2017.

[89] Frank Sup, Huseyin Atakan Varol, and Michael Goldfarb. Upslope walking with a powered knee and ankle prosthesis: initial results with an amputee subject. *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, 19(1):71–78, 2011.

[90] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic robotics*. MIT press, 2005.

[91] D. C. Tkach and L. J. Hargrove. Neuromechanical sensor fusion yields highest accuracies in predicting ambulation mode transitions for trans-tibial amputees. In *2013 35th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 3074–3077, 2013.

[92] Brigitte Toro, Christopher J Nester, and Pauline C Farren. Cluster analysis for the extraction of sagittal gait patterns in children with cerebral palsy. *Gait & posture*, 25(2):157–165, 2007.

[93] Lisa Torrey and Jude Shavlik. Transfer learning. In *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques*, pages 242–264. IGI global, 2010.

[94] Matthew Trumble, Andrew Gilbert, Charles Malleson, Adrian Hilton, and John Collomosse. Total capture: 3d human pose estimation fusing video and inertial sensors. In *Proceedings of 28th British Machine Vision Conference*, pages 1–13, 2017.

[95] Michael Tschiedel, Michael Friedrich Russold, and Eugenijus Kaniusas. Relying on more sense for enhancing lower limb prostheses control: a review. *Journal of Neuro-Engineering and Rehabilitation*, 17(1):1–13, 2020.

[96] Michael R Tucker, Jeremy Olivier, Anna Pagel, Hannes Bleuler, Mohamed Bouri, Olivier Lambercy, José del R Millán, Robert Riener, Heike Vallery, and Roger Gassert. Control strategies for active lower extremity prosthetics and orthotics: a review. *Journal of neuroengineering and rehabilitation*, 12(1):1, 2015.

[97] Heike Vallery, Rainer Burgkart, Cornelia Hartmann, Jürgen Mitternacht, Robert Riener, and Martin Buss. Complementary limb motion estimation for the control of active knee prostheses. *Biomedizinische Technik/Biomedical Engineering*, 56(1):45–51, 2011.

[98] Vassilios G Vardaxis, Paul Allard, Régis Lachance, and Morris Duhaime. Classification of able-bodied gait using 3-d muscle powers. *Human Movement Science*, 17(1):121–136, 1998.

[99] Huseyin Atakan Varol, Frank Sup, and Michael Goldfarb. Multiclass real-time intent recognition of a powered lower limb prosthesis. *Biomedical Engineering, IEEE Transactions on*, 57(3):542–551, 2010.

[100] Eric Watelain, Franck Barbier, Paul Allard, André Thevenon, and Jean-Claude Angué. Gait pattern classification of healthy elderly men based on biomechanical data. *Archives of physical medicine and rehabilitation*, 81(5):579–586, 2000.

[101] Yue Wen, Jennie Si, Andrea Brandt, Xiang Gao, and He Huang. Online reinforcement learning control for the personalization of a robotic knee prosthesis. *IEEE transactions on cybernetics*, 2019.

[102] Jason M Wilken, Kelly M Rodriguez, Melissa Brawner, and Benjamin J Darter. Reliability and minimal detectible change values for gait kinematics and kinetics in healthy adults. *Gait & posture*, 35(2):301–307, 2012.

[103] David A Winter. *Biomechanics and motor control of human gait: normal, elderly and pathological*. 1991.

[104] Erik J Wolf, Vanessa Q Everding, Alison L Linberg, Barri L Schnall, Joseph M Czerniecki, and Jeffrey M Gambel. Assessment of transfemoral amputees using c-leg and power knee for ascending and descending inclines and steps. *Journal of Rehabilitation Research & Development*, 49(6), 2012.

[105] Ge Wu, Sorin Siegler, Paul Allard, Chris Kirtley, Alberto Leardini, Dieter Rosenbaum, Mike Whittle, Darryl D D'Lima, Luca Cristofolini, Hartmut Witte, 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. *Journal of biomechanics*, 35(4):543–548, 2002.

[106] Spyros Xanthopoulos and Stelios Xinogalos. A review on location based services for mobile games. In *Proceedings of the 20th Pan-Hellenic Conference on Informatics*, page 28. ACM, 2016.

[107] Tingfang Yan, Yuxiang Sun, Tingting Liu, Chi-Hong Cheung, and Max Qing-Hu Meng. A locomotion recognition system using depth images. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6766–6772. IEEE, 2018.

[108] Jianbo Yang, Minh Nhut Nguyen, Phyo Phyo San, Xiaoli Li, and Shonali Krishnaswamy. Deep convolutional neural networks on multichannel time series for human activity recognition. In *Ijcai*, volume 15, pages 3995–4001, 2015.

[109] A. J. Young and L. J. Hargrove. A classification method for user-independent intent recognition for transfemoral amputees using powered lower limb prostheses. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 24(2):217–225, 2016.

[110] Aaron J Young, Ann Simon, and Levi J Hargrove. An intent recognition strategy for transfemoral amputee ambulation across different locomotion modes. In *Engineering in medicine and biology society (EMBC), 2013 35th annual international conference of the IEEE*, pages 1587–1590. IEEE, 2013.

[111] Aaron J Young, Ann M Simon, Nicholas P Fey, and Levi J Hargrove. Classifying the intent of novel users during human locomotion using powered lower limb prostheses. In *Neural engineering (NER), 2013 6th international IEEE/EMBS conference on*, pages 311–314. IEEE, 2013.

[112] Fan Zhang, Zheng Fang, Ming Liu, and He Huang. Preliminary design of a terrain recognition system. In *Engineering in medicine and biology society, EMBC, 2011 annual international conference of the IEEE*, pages 5452–5455. IEEE, 2011.

[113] Kuangen Zhang, Clarence W de Silva, and Chenglong Fu. Sensor fusion for predictive control of human-prosthesis-environment dynamics in assistive walking: A survey. *arXiv preprint arXiv:1903.07674*, 2019.

[114] Kuangen Zhang, Caihua Xiong, Wen Zhang, Haiyuan Liu, Daoyuan Lai, Yiming Rong, and Chenglong Fu. Environmental features recognition for lower limb prostheses toward

predictive walking. *IEEE transactions on neural systems and rehabilitation engineering*, 27(3):465–476, 2019.

[115] E. Zheng, Q. Wang, and H. Qiao. An automatic labeling strategy for locomotion mode recognition with robotic transtibial prosthesis. In *2019 IEEE 9th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER)*, pages 1010–1013, 2019.

[116] Enhao Zheng, Long Wang, Kunlin Wei, and Qining Wang. A noncontact capacitive sensing system for recognizing locomotion modes of transtibial amputees. *IEEE Transactions on Biomedical Engineering*, 61(12):2911–2920, 2014.

[117] Boxuan Zhong, Rafael Luiz da Silva, Minhan Li, He Huang, and Edgar Lobaton. Environmental context prediction for lower limb prostheses with uncertainty quantification. *IEEE Transactions on Automation Science and Engineering*, 2020.

VITA

Vijeth Rai is wandering student of life. Through technology, he seeks to enable mobility for all to wander this beautiful world.