Using wearable sensors and machine learning to enrich lower limb assistive devices

Abhishek Sharma

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

Doctor of Philosophy

University of Washington

2023

Reading Committee:
Eric Rombokas, Chair
Kat Steele
Sawyer Fuller

Program Authorized to Offer Degree:
Mechanical Engineering

University of Washington

Abstract

Using wearable sensors and machine learning
to enrich lower limb assistive devices

Abhishek Sharma

Chair of the Supervisory Committee:
Eric Rombokas
Mechanical Engineering

In the past, our understanding of how humans move has been greatly restricted because of the tools available to us. Firstly, we had limited gait datasets, which mostly captured simple activities like walking in a straight line at a steady pace or walking on a treadmill. Interesting activities like dancing or obstacle avoidance were largely confined to laboratories, which did not necessarily reflect the complexities and variations that may arise from interactions in the real world. Secondly, gait modeling tools were also limited. So, modeling techniques had to make strong assumptions about human gait like we use cyclic repetitive movement of limbs or that we select from a finite repertoire of activities like flatground walking, stair ascent, descent etc. This has greatly reflected in how we design control for our assistive devices. Thus, most existing control strategies select from a finite list of activities with corresponding predefined gait profiles, which are then repeated in a periodic manner. Even for common activities like flatground walking slight variations that might arises due to interactions with the environment like going around an obstacle are largely ignored. The variations across people are also not accounted for when controlling assistive devices due to extensive tuning required. This dissertation aims at proposing solutions to these challenges by using recent advances in wearable sensors and machine learning. In recent times, there have been great improvements in wearable motion capture systems as well as wearable cameras, and these are expected

to grow further in the next few years. This allows human movement tracking in real world environments, as well as recording how humans react to the objects in their environment, and navigate around them. These large datasets then allow the use of data-driven modeling techniques to better understand human movement, and improve assistive devices. We combine the technological improvements in wearable sensing, with improvements in data-driven modeling to address the 4 key current limitations. 1) We tackle the problem of limited gait models in coordinated movement, where we propose an **expressive predictive model** of gait using recurrent neural networks, which can predict gait for a variety of activities, and does not rely on assumptions about periodicity of gait, or a clearly defined finite set of activities. 2) We analyze how gait predictability is affected by **environments**, and investigate the use of environment information in improving the predictive models of gait. 3) We investigate how to quantify and compare the **complexity** of different activities, and provide practical recommendations about the usage of different measures of complexity. 4) We develop a style transfer based solution to **personalize assistive** devices, with the aim of reducing the effort required for manual tuning, and also propose how to quantify personal style of a user. The dataset generated as a part of this dissertation is freely available and can be accessed through the following github repository- https://github.com/abs711/The-way-of-the-future

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGMENTS

I wish to express my sincere gratitude to my PhD advisor and mentor Dr. Eric Rombokas. Thank you for letting me explore some of the craziest ideas and projects, and helping me make them more concrete. Thank you for being patient, and resilient during crazy times in the last 3 years, and for creating the safe space for me to focus only on my research.

I would like to thank Vijeth for expediting the learning process during the early phase of my PhD. I developed a lot of skills by reviewing and understanding the codebase your created, and a significant portion of my work builds upon what you did during your PhD.

I would like to thank the members of my supervisory committee, Dr. Kat Steele, Dr. Sawyer Fuller, and Dr. Sam Burden, for their valuable insight, discussions, and questions which I still think about, and which would hopefully lead to something fruitful in the future.

I would like to thank David and Astrini, for letting me be free of the administrative hassles, and offering their kind advice on academic as well as post-academic matters. Also, thanks Jom for being the coolest person I know. I admire your work as an engineer and a designer, and also relate with your aversion to humans. JK. I would also like to thank Maxim for helping me polish my presentations, and providing valuable feedback on manuscripts. I also find your conscientiousness, creativity, and curiosity to learn very inspiring.

I would like to thank Wanwisa for lighting up the department amidst all the doom and gloom, for all the candies I stole from you, and for offering guidance even though I'd drop in without scheduling appointments. I would also like to thank Maria for help with my defense practice and countless assignments. Have lost count of the coffees I owe you.

Finally, I would like to thank Ketan, Nicole, Sayem, *Chinks*, and the duo Parth and Pranjal, for being my friends and mentors. This would be unimaginable without you all.

DEDICATION

To my parents, and Man$u$;)

&

To anyone who wants to build upon this work

Chapter 1

INTRODUCTION

1.1 Motivation

Natural bipedal locomotion observed in humans is capable of a diverse set of movements, and maneuvers with agility that is rarely witnessed in any other system- natural or artificial. This ability enables humans to adapt to various situations and terrains, as well as perform feats like dance, climb, leap, skate etc. The freedom, a human can experience within the bounds of physics is enormous and unrivaled. Therefore, loss of limb can lead to equally significant reduction in ability, agency and quality of living of a human being.

Powered lower-limb assistive devices such as prostheses and orthoses can improve the quality of life of individuals with affected limbs, by providing support, a sense of stability and net positive mechanical work. Positive work enables users to perform various activities without significant compensation, reduces effort and metabolic cost [8, 66].

Even though these devices are beneficial, they are limited in many ways in their current form. The strategy for control of assistive devices is built upon our current understanding of human locomotion and gait analysis [49]. However, the tools of movement analysis that were available to us in the past have been technologically constrained.

Firstly, there has been a limited amount of gait data, which mostly captured activities like flatground walking in a straight line or walking on a treadmill [72, 38, 37, 13, 100, 44], due to cumbersome equipment involved in marker-based motion capture systems. More complex activities exist in smaller amounts (e.g. less than a minute of data from a single person walking around an obstacle) and are captured within the confines of a laboratory [1, 12], using marker-based motion capture systems. However, this data do not necessarily reflect the variety and rich interactions with the environment that might occur in the real world. In

addition, there is evidence that artificial environments can compromise ecological validity. Friesen et al. [36] showed that how participants walk changed by a small but significant amount depending on the number of researchers present during the experiment.

Secondly, the gait modeling techniques have also been limited, necessitating the use of models which make strong assumptions about human gait like cyclic repetitive movement of limbs or that humans only select from a finite cleanly defined repertoire of activities like flatground walking, stair ascent and descent. This is also reflected in how control is designed for lower-limb assistive devices. Control of assistive devices requires the ability to predict the user-intent, and their needs given the task they are performing and the environment they are in. This require a predictive model of the gait. Most models used in control of assistive devices are designed to perform a finite set of activities and predefined profiles (joint angles, torques, impedances) for each activity.

These limits on the motion capture systems, and gait modeling techniques have restricted human movement analysis and consequently the current capabilities of assistive devices in 4 important ways. We have assistive devices that 1) are capable of performing a limited set of activities, 2) require distinct controllers for different activities, each of which needs manual tuning, 3) do not utilize the rich information available from the environment about the user's future behavior, 4) are generic or manually personalized to do a single task. In our work, we show that combining technological improvements in portable wearable motion capture systems and wearable cameras, with improvements in data-driven modeling, can be used to improve assistive devices by enabling, more expressive predictive modeling of gait that can scale to a variety of activities, utilization of rich information from the environment to improve the predictive models of gait, and personalization of assistive devices to account for interpersonal differences.

1.2 Research aims

For assistive devices to achieve the degree of complexity displayed by natural human gait, we need developments in the following 4 directions:

Aim 1: Expressive predictive modelling of gait using ***Coordinated Movement***

In order to tackle the limitations of current gait modeling approaches that make strong assumptions about human gait, like periodicity of gait and clearly defined activities, we need to investigate more expressive ways of modeling gait, which don't rely on segmentation of gait. Coordinated movement is our proposed solution to this problems, using recurrent neural networks, which we investigate in chapter 2. Chapter 2 presents the key findings from the following 3 publications:

- Mode-free control of prosthetic lower limbs [83]
- Continuous and unified modelling of joint kinematics for multiple activities [85]
- Coordinated movement for prosthesis reference trajectory generation: Temporal factors and attention [86]

Aim 2: ***Environment modeling for gait prediction using egocentric vision***

To address the limited understanding of human gait in natural environment, we collect locomotion data in natural environments [102] and investigate how environments (particularly congested or open spaces) affect gait predictability in Chapter 3. Particularly, we investigate whether information from the environment can be used to improve the predictive models of gait that we develop in chapter 2: Coordinate Movements. The chapter presents key finding from the following publication:

- Improving imu-based prediction of lower limb kinematics in natural environments using egocentric optical flow [104]

Aim 3: Understanding the ***Complexity of locomotion activities***

To understand how to quantify the complexity of human locomotion, in Chapter 4 we investigate several commonly used measures in gait literature. We show that the most commonly

analyzed activity in the gait literature i.e. flatground walking in a straight line, is also the simplest, and present grounds for investigating a greater variety of locomotion activities, and how to compare different activities in terms of their complexity. This chapter draws from the following publication:

- Complexity of locomotion activities in an outside-of-the-lab wearable motion capture dataset[103]

Aim 4: ***Optimizing representations of multiple simultaneous attributes for personalized gait generation using deep learning***

To solve the challenge of incorporating interpersonal differences into assistive device control, in Chapter 5 we investigate how to quantify the personal style of a user, and how to generate gait trajectories specific to them. These trajectories can be used to control the assistive devices. This chapter draws from the following publication in review:

- Optimizing representations of multiple simultaneous attributes for personalized gait generation using deep learning (in review)

Chapter 2

COORDINATED MOVEMENT

Vijeth Rai, Abhishek Sharma, David Boe, Pornthep Preechayasomboon, Eric Rombokas

IEEE Access, 2022 [85]

Abstract

Intuitive control of powered prosthetic lower limbs is still an open-ended research goal. Current controllers employ discrete locomotion modes for well-defined and frequently encountered scenarios such as stair ascent, stair descent, or ramps. Non-standard movements such as **side-shuffling into cars and avoiding obstacles** are challenging to powered limb users. Human locomotion is a continuous motion comprising rhythmic and non-rhythmic movements, fluidly adapting to the environment. It exhibits strong inter-joint coordination and the movement of a single joint can be largely predicted based on the movement of the rest of the body. We explore a continuous and unified kinematics estimation strategy for a wide variety of movements without the need for labeled examples. Our data-driven approach uses natural body motion from the intact limbs and trunk to generate a kinematic reference trajectory for prosthetic joints. Wearable sensors were worn by 63 subjects without disabilities to record full-body kinematics during typical scenarios (flat ground and stairs), and non-rhythmic and atypical movements (side shuffles, weaving through cones, backward walking). A Recurrent Neural Network (RNN) was trained to predict right ankle and knee kinematics from the kinematics of other joints as inputs. Results were assessed on 3 different test subjects previously unseen by the network. All predictions had a RMSE of less than 7.5 degrees and a high correlation across activities. These offline predictions were robust to subject-specific variations such as walking speed and step length. Additionally, to test the

feasibility of using a data-driven reference towards prosthetic control in real-time, a systems test was designed with a single participant. The controller acquired live kinematics, generated predictions using a pre-trained neural network, and demonstrated the capability to actuate the knee joint of a powered prosthesis for the treadmill walking task.

2.1 Introduction

In this chapter, we present a data-driven framework for gait prediction, that could serve as a reference trajectory generator for the control of assistive devices. The goal is to build a predictive model of gait that eliminates the requirement of discretizing human gait into distinct activities and phases, that is currently the standard in current powered prosthesis limbs. We present a unified approach to predicting joint trajectories continuously (without splitting into distinct phases or activities) for a variety of human locomotion activities. This minimizes the need for deciding conditions to switch between distinct phase and activity controllers, and hand-tuning of gait parameters for the control of assistive devices.

Next, we discuss the motivation for this work. In section 2.2, the methods are presented. Section 2.3 presents the key results, and finally the limitations and challenges are discussed in 2.4.

2.1.1 Motivation

Traditional methods for control of assistive devices use a hierarchical 3-level architecture as shown in Fig. 2.1. The high-level controller uses data from sensors on the device like goniometer, imu etc. to determine the current activity mode (e.g. flatground walking), and whether a mode switch (e.g. flatground to stair ascent) is about to happen. The decision from the high-level controller is fed into the mid-level controller which detects gait events like heel-strike or toe-off and triggers the predefined trajectory for the next phase until the next gait event is detected. Both high-level and mid-level controllers are implemented using Finite State Machines (FSM) (Figs. 2.2a. and 2.2b. respectively). The reference trajectory for joint angles, torques or impedence determined by the mid-level controller are sent to the low-level controller which executes the desired command on the hardware.

This FSM-based control architecture requires several controllers to encode the continuum of all tasks and phases. With increase in the number of controllers, the number of parameters to tune can combinatorially explode. This increases the effort and time to tune the device

to the user [106]. This discrete approach to control requires very precise prediction of phase and activity. Slight error in activity or phase detection can lead to falls. In addition, once a state (activity and phase) has been decided it has to be completed, and the user cannot abruptly change gait in the middle of a phase.

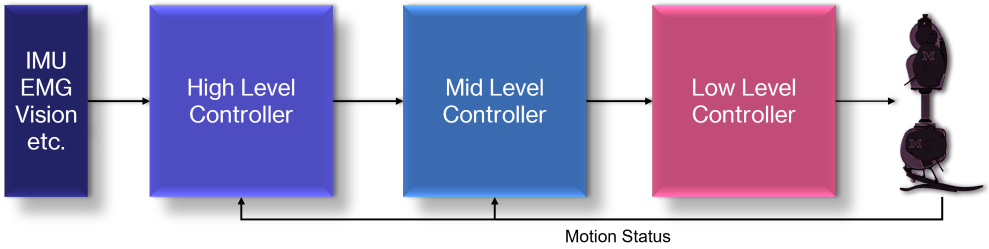

Figure 2.1: **Hierarchical 3-level control architecture**, adapted from [119]

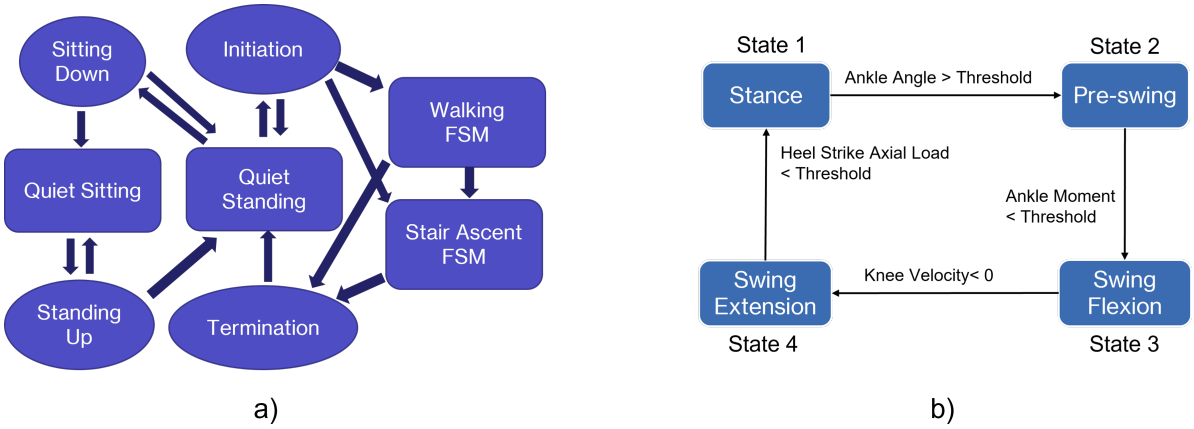

Figure 2.2: A FSM-based high-level controller adapted from [77] (left), and a mid-level controller for flatground walking adapted from [113](right)

An alternative approach to controller design is to model gait continuously without splitting into discrete sections. This approach known as *Continuous Control*, models affected limb trajectory as a continuous function of intact limbs [82]. These approaches use models

trained on gait data collected from non-disabled adults. For example, [31] model knee joint as a function of thigh angle in the sagittal plane using Gaussian Process Regression trained on data from intact population walking on a treadmill. The predicted joints then serve as reference to the low-level controller (See Fig. 2.3).

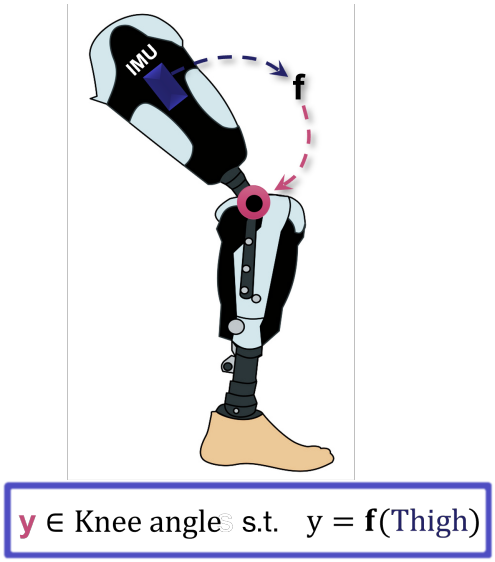

Figure 2.3: **Continuous Control:** The missing joint trajectory is modelled as a continuous function of the intact limb data, without explicitly splitting gait into distant phases.

Most existing work on continuous control [82, 31, 120] only combines the discrete phases of gait into a continuous model. A high-level controller is still required to decide the activity mode. Exceptions are [29, 90], where the authors have modelled gait continuously over a range of slopes and speeds, from data collected during a treadmill walking task.

In this study, we introduce the following changes from the previous approaches: 1) Collect data using wearable motion capture [2] in real world environments, 2) Collect data for activities that would be difficult to categorize into modes, 3) Build a unified gait prediction model that can represent multiple activities and predict missing joint kinematics, which can serve as reference for control of assitive devices. 4) Investigate advantages of using full body kine-

matics data for prediction as compared to only lower-limb data as inputs to the model.

2.2 Methods

2.2.1 Participants

Ambulation data was collected for a total of 63 non-disabled participants (34 male, median age 25). Recruitment and human subject protocols were performed in accordance with the University of Washington Institutional Review Board guidelines.

2.2.2 Activities

The movements targeted in this study were designed to be more challenging 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 See Fig. 2.4. 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. See Fig. 2.4

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.

Figure 2.4: Corridor for flat ground walking (left), and staircases spanning 6 floors (right)

They include challenging abrupt changes in the movement direction, running, and backward locomotion. These standard tests are described in detail in [39, 87] 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 test aims to complete a weaving running course in the shortest possible time. The obstacle course is depicted in Fig. 2.5. 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 rightmost cones as well as

backward walking to return to the starting location.

The distribution of subjects across activties is shown in Table 2.1.

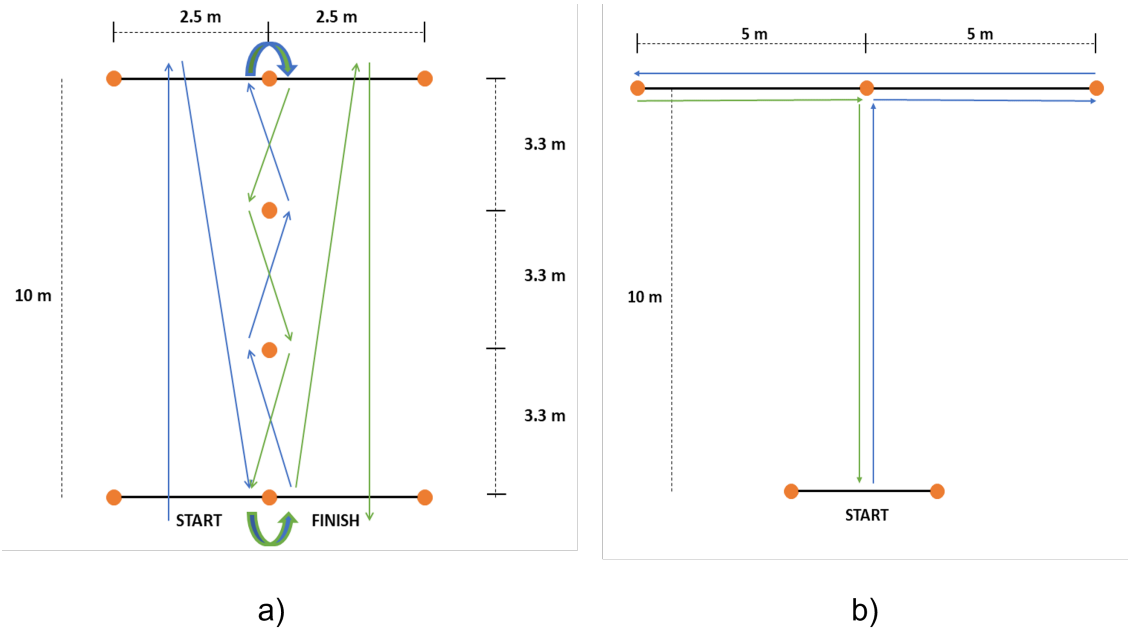

Figure 2.5: Obstacle course for CHAMP activities

Group	***Activity***	***Number***
Group A	Flatground walking with random stops	11 subjects
Group B	CHAMP Tests	10 subjects
Group C	Stair ascent & descent	42 subjects

Table 2.1: Distribution of subjects across the 3 different activities. Since, data collection is an inefficient process (due to recalibrations required to maintain data quality) with effectively 25 minutes of data available from a 2 hour session, not all the subjects performed every activity.

2.2.3 Data Collection

Locomotion data were collected using the Xsens Awinda suit [2], consisting of 17 body-worn sensors placed at each segment of the limbs, as well as sternum, sacrum, shoulder scapula, and forehead (See the xsens user manual [5]). After a system specified calibration, the software provides joint kinematics in a 3D environment. 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 [126]. Data sampled at 60 Hz, from a total of 22 joints in 3 anatomical planes (sagittal, frontal, transverse) was captured for each trial.

For each data collection session, the subject was briefed about the task to be performed and the equipment to be donned, and informed consent was obtained. Subjects first performed a standard n-pose calibration routine as dictated by the device manufacturer. Following this, the data collection began. 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 a self-selected pace. Participants were encouraged to take breaks after each trial. Duration of the entire setup and data collection was around 1.5 - 2 hours. In total, 750 minutes of data were collected from all subjects .

2.2.4 Data Processing

Sensor data were 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 op-

timal 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 a significantly longer time (anywhere between a few minutes to a few hours depending on the size of the files).

2.2.5 Machine Learning Details

Our goal was to build a predictive model that predicts missing limb trajectory from intact limb motion. In [83], we demonstrated the usefulness of including time-history of intact limb joint angles in predicting missing right ankle joint. We found that a Recurrent Neural Network - Long Short-Term Memory (LSTM) [43] was better than Multivariate Linear Regression on instantaneous data and a Full Connected non-recurrent network trained on time history data. In [85] we increased the number of joints predicted to two by including the right knee joint as well. The system architecture for gait predicition is shown in Fig. 3.4 The deep-learning networks were implemented on Tensorflow for [83] and PyTorch for [85].

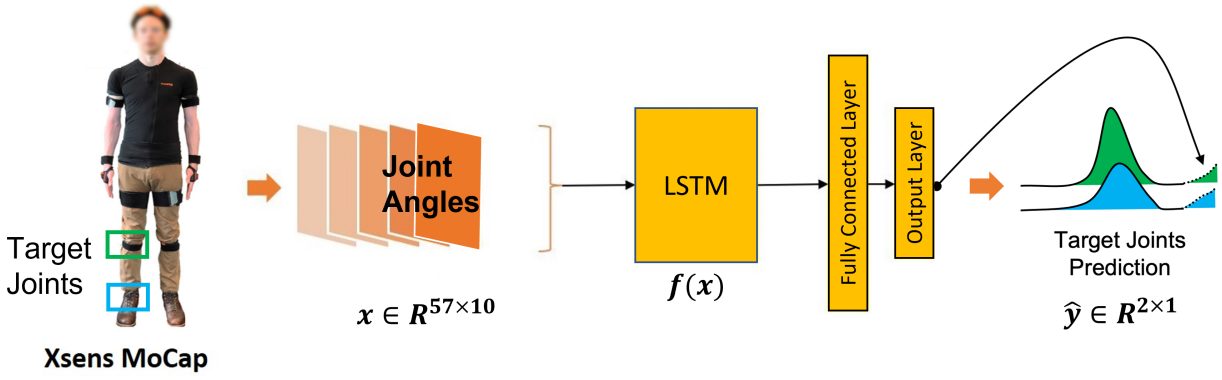

Figure 2.6: **Architecture for joint trajectory prediction**

In the real-time prosthetic controller, the trained network would be applied to predict joint trajectories for a user whose movements are not captured in the training dataset. Thus, the subjects were randomly split into training, validation, and test with no overlapping

	Flatground	**CHAMP**	**Stairs**
Train	9 subjects Group A	8 subjects Group B	40 subjects Group C
Test	1 subject Group A	1 subject Group B	1 subject Group C
Validation	1 subject Group A	1 subject Group B	1 subject Group C

Table 2.2: Distribution of subject data for training, test and validation sets for each of the cross validation runs. This process was repeated thrice with random shuffling and the average result across the 3 runs was reported.

subject data. For the offline tests, we used 3 runs of leave-one-out cross-validation method to report generalized results. Table 2.2 shows the distribution of subjects. The validation data was used to tune the hyperparameters and the test set was used to report performance for that particular run. This process was repeated thrice with a random allotment of subjects as training, test, and validation data. For each run, 57 subjects were included in training set, 3 in test and 3 in validation set. The final results reported were averaged across the three runs. The data for the test (and validation) set was combined from 3 test subjects such that all 3 activity types were included.

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.

$$\hat{y_T} = f(x) \tag{2.1}$$

where f is the LSTM network.

Data Normalization and Reshaping

Each of the joint angles exhibits a different Range of Motion (ROM). 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.

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 [88]. Training input samples were prepared as an overlapping rolling window of time series data of desired sequence length. The optimal sequence length was a hyperparameter we tuned for using grid search (See table 3.1).

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 a common metric used for regression tasks in machine learning. Apart from the sequence length, the network also has several hyperparameters 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

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 2.3: Hyperparameter values tested for optimal performance on Obstacle course dataset

regularization parameter for L2 loss (λ), and learning rate.

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 3.1. The optimal hyperparameter set was used to compare and evaluate performance.

Denormalization

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

Evaluation Metrics

We used 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.

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.

2.3 Results and discussion

In this section, we present key results from three publications [83, 86, 85] , towards development of our unified expressive gait predictor.

2.3.1 Including time history aids prediction performance

In [83], we showed that including the time history of intact limb as inputs to the model helped significantly improve the right ankle prediction. Inspired from [120], we used a Multivariate Linear Regression model on our data as one the baselines. The model predicted the right ankle joint at the next instant with instantaneous data (No time history) from rest of the joints. We also used a full connected deep neural network (DNN) as the second baseline. The DNN was fed a fixed window of time history as the input. The LSTM was the best performing model as shown in Fig. 2.7. Thus, we use LSTM for the rest of our work in gait modelling.

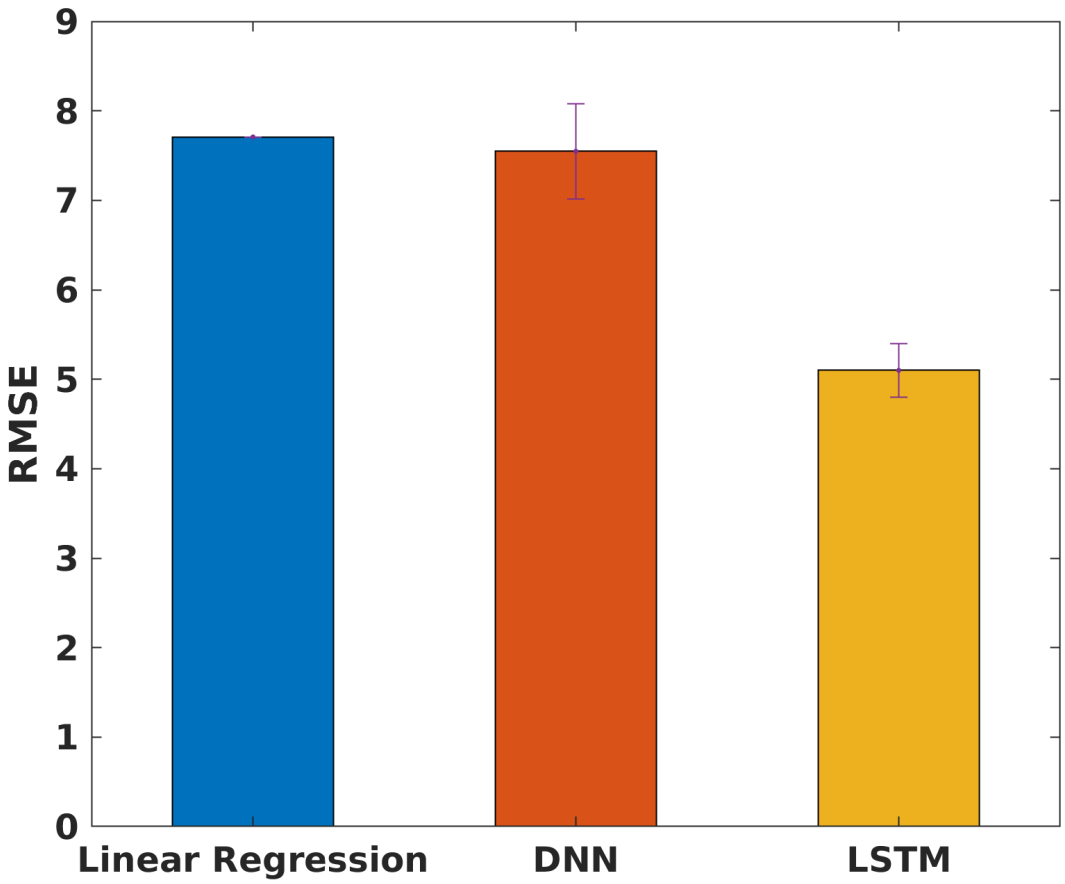

Figure 2.7: Error for each class of machine learning algorithm. All networks predicted the right ankle angle from the time-history of remaining joints. Both neural network models showed improved performance, with the Recurrent Neural Network (LSTM) performing best by a large margin [83].

2.3.2 Data from the frontal and transverse planes aids performance

In Fig 2.8, we see that using data from the frontal and transverse plane reduces the RMSE for ankle prediction during stair ascent and descent tasks. The performance of the models trained on Full Body input and only Lower-limb input is similar.

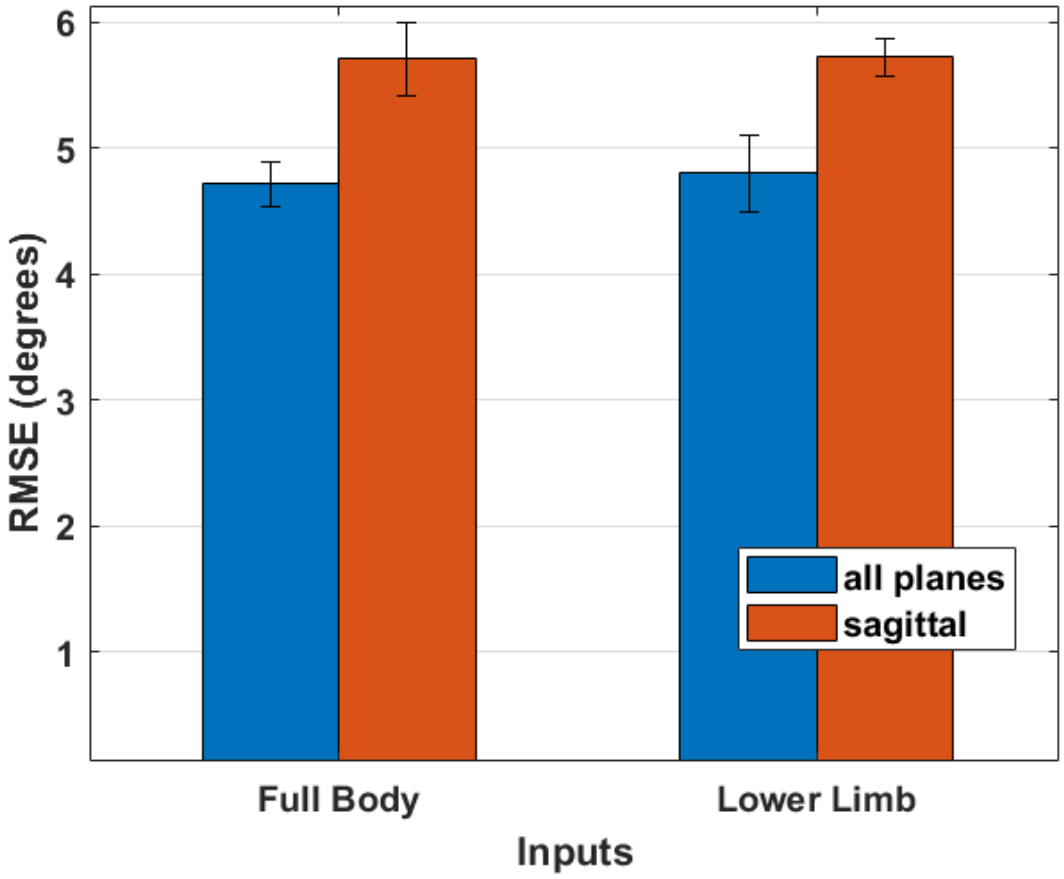

Figure 2.8: Performance for subsets of sensors. The LSTM network predicted the right ankle angle from the time-history of remaining joints in full body and only lower limb. Considering data from all 3 anatomical planes (Frontal, Transverse and Sagittal) improved performance. Using only the lower limb sensors for training did not result in significant drop in performance [83].

2.3.3 A unified model that can predict knee and ankle joints for a variety of activities

In [85], we extended the LSTM model to include all the activities and predict both knee and ankle. Fig. 2.9 shows the trajectories generated by the network for different activities over a 3 second window.

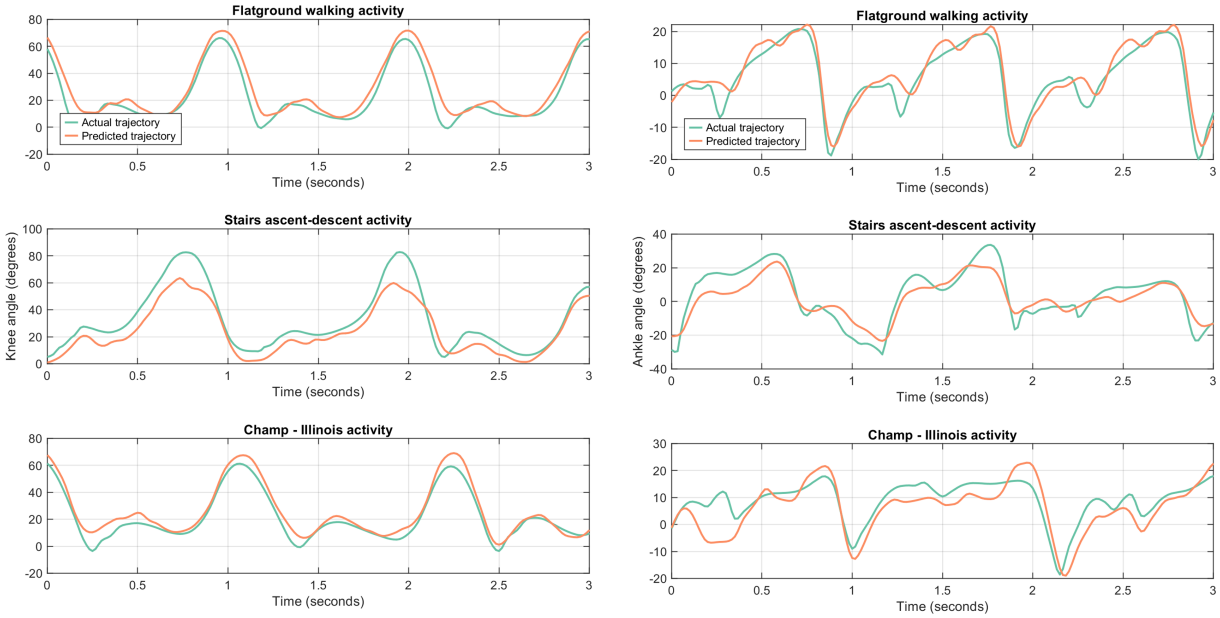

Figure 2.9: Knee (left) and Ankle (right) joint predictions for 3 different activities generated by the same network.The trajectories shown for a test subject whose data was not part of the training data. About 3 seconds of actual measured (green) and predicted (red) trajectory for flat-ground (top), 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.

Fig. 2.10 shows the performance comparison for different activities. Performance decreased as the complexity of activities increased. Table 2.4 shows the Pearson Correlation Coefficient for knee and ankle predictions. Table 2.5 shows the prediction error as percentage of range of motion.

The CHAMP activity predictions had reduced performance for both joints, suggesting that the complexity of movements involved in an activity impacts prediction performance. The RMS error of predictions was generally within 12% of the range of motion for each

Table 2.4: 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**	
	mean	**std**	**mean**	**std**
Flat-ground	0.91	0.024	0.97	0.0067
Stairs	0.86	0.070	0.96	0.0015
CHAMP	0.72	0.066	0.88	0.026

Table 2.5: RMS errors of predictions as percentages of range of motion. The CHAMP activity had the highest error percent for both joints.

	Ankle			**Knee**		
	Error (degs)	**ROM (degs)**	**Error (%)**	**Error (degs)**	**ROM (degs)**	**Error (%)**
Flat ground	5.6	50.3	11.2	3.6	77.15	4.7
Stairs	4.7	78.2	6.0	4.4	99.59	4.4
CHAMP	7.5	46.4	16.3	6.7	70.34	9.5

activity (Table 2.5). The only exception was the ankle joint predictions for CHAMP activities with about 16.3%. Similarly, the ankle joint predictions for CHAMP activities (Table 2.4) had the lowest correlation of 0.72. 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 a similar benefit to performance for CHAMP activity as we discuss in the following section.

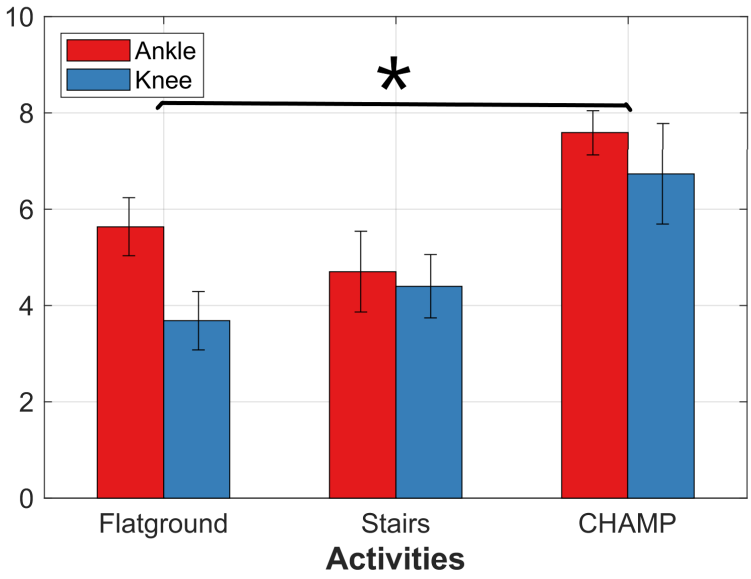

Figure 2.10: RMS error with respect to individual activities for ankle joint (red) and knee joint (blue) sagittal plane predictions. Performance was within 8 degrees RMS error for all activities and both joints. Statistical analysis showed that complexity of activity significantly increased RMS Error.

2.3.4 Predicting transitions between different activities

Fig 2.11 shows an example of a smooth transition from stair descent to flatground. A complete rotation executed during the Illinois Agility Test (IAT) and transition from side-stepping to backward walking executed during T-test activity is shown in Fig. 2.12. All transitions were predicted successfully and instantaneously, however, performance varied across different types of activities. While the more structured activity of stair descent had an RMS error of 2.68 degrees, the CHAMP transitions showed a marked increase in the error of 5.9 and 7.6 degrees respectively.

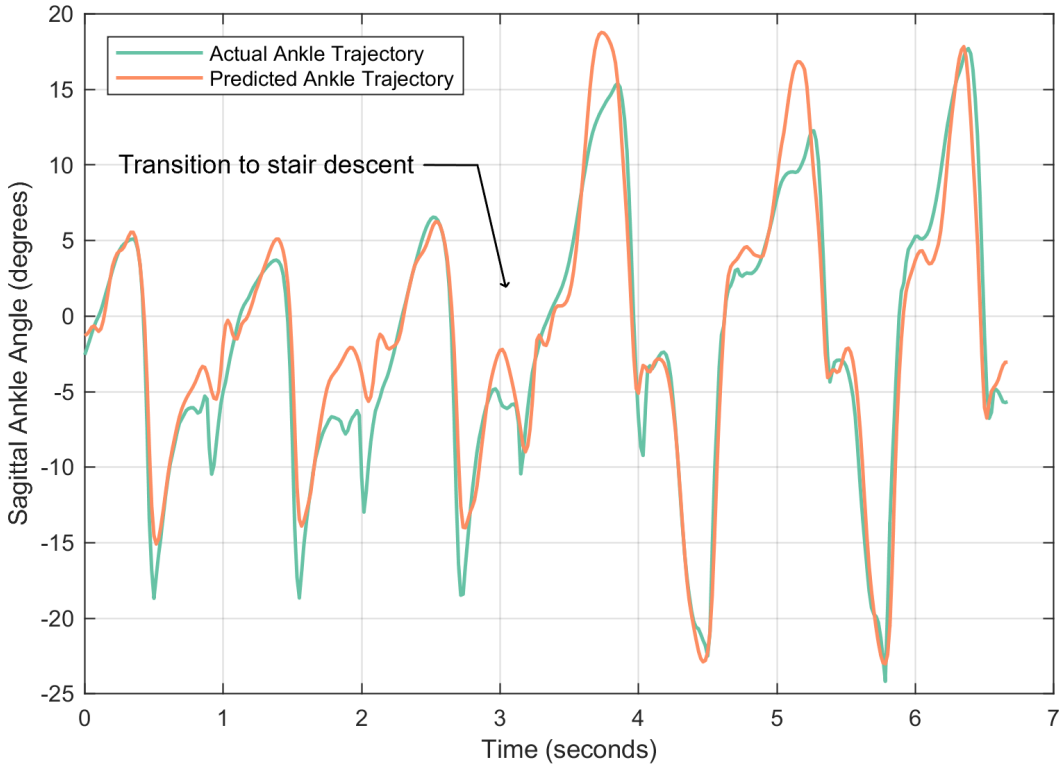

Figure 2.11: Ankle joint predictions showing continuous seamless transition from flat ground walking to stair descent.

2.3.5 Performance improves with more data

Fig. 2.13 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. This is an important result suggesting that the performance of hard activities (like CHAMP) 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. This could be because adding more data from the same subject does not add any new variation to the overall repertoire of movements seen by the network.

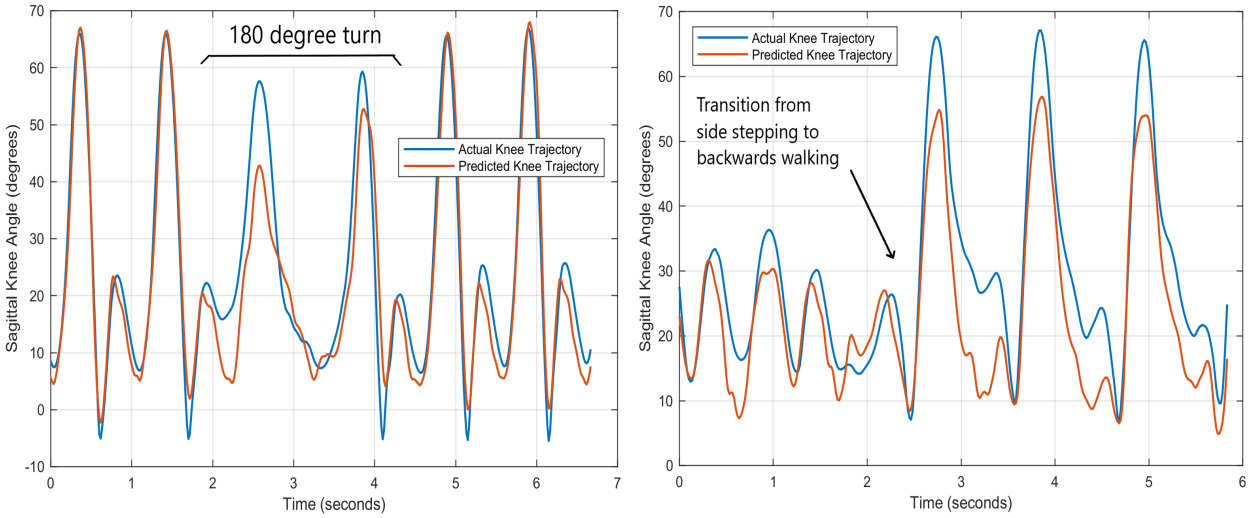

Figure 2.12: Knee joint predictions showing continuous seamless transitions in 180-degree rotation executed during the Illinois Agility Test (left) and from side-stepping movement to backwards walking in T-test (right).

2.4 Limitations and challenges

2.4.1 Use of RMSE and PCC as a metric

Since our method involves regressing to target joint angles, we have to use rmse as the performance metric. While traditional FSM based controllers report mode-classification accuracy for performance. Thus, comparison with those methods is difficult.

RMSE has been used in continuous direct myoelectric control studies to report performance. For the case of flatground walking, we see [32] and [24] report a RMSE of around 6 degrees and 7.5 degrees in the sagittal plane for ankle joint kinematics. In comparison, we report an error of 5.6 degrees for the same. [50] used Gaussian Process Regression to learn continuous knee joint kinematics from spatio-temporal ultrasound features from leg muscles . They report a RMSE of 4.7 and 10.7 degrees for flatground walking and stair ascent activity respectively. In comparison, we observe an RMSE of 3.6 and 4.4 degrees for

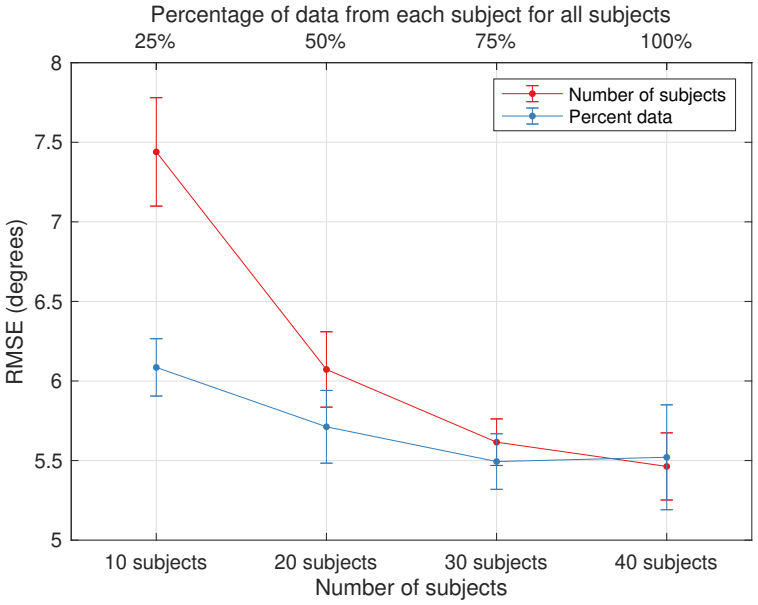

Figure 2.13: RMS error with respect to the number of subjects (bottom X-axis) and percentage of data used from all subjects (top X-axis) for stair ascent and descent data. More subjects included in training data resulted in a statistically significant performance gain. Keeping the total number of subjects the same (n=40), but using only 50% of the data showed approximately the same performance.

the same. However for the CHAMP activity, there are no results to compare with in the existing literature. Similarly we find that the PCC for knee prediction during flatground walking is 0.97 which is comparable to [89], who observe PCC ranging from 0.8-0.93, for flatground walking at various speed.

Another, concern is that metrics like RMSE and PCC do not necessarily reflect the subjective experience of the user. [129] investigated the effects of errors in locomotion mode classification on user perception of instability. The study evidenced that the timing with respect to the gait phase of the control disturbances caused due to errors is an important factor. So, for our method another way to look at error could be to look at the maximum stance-phase error. For example, for flat-ground walking, the average RMSE for ankle and

knee were 5.6 and 3.6 degrees, and the average max error during the stance phase was 6.77 and 8.99 degrees, across different gait cycles.

2.4.2 Use of RMSE as a Loss function

The max error (ankle: 6.77 and knee: 8.99 degrees) reported above being much larger than the RMSE (ankle: 5.6 and knee: 3.6 degrees) highlights another challenge associated with training our gait model using RMSE as a loss function. The models don't emphasize on the criticality of events like stance which might be relevant to human subjective experience, and treat all the moments in the gait as same. Thus, we need a loss function that can account for this fact and place greater emphasis on events critical to smooth user experience.

2.4.3 Rehabilitation vs Accessibility

Predicting the movements of an able-bodied population is only the first step toward using those predictions to create intelligent prosthesis control. Movements generated by able-bodied users almost certainly differ from those generated by prosthesis users. This gets at the heart of a central problem in rehabilitation training. What is the right gait that we should strive for? Should individuals undergoing rehabilitation be encouraged to walk exactly like able-bodied individuals (Rehabilitation), or since the prosthesis and amputation affect dynamics, should the gait be adapted to maximise the user's potential? (Accessibility) In the former case significant training at the user end must be designed so that they can explore the behavior of the prosthesis and train themselves to use it. In the latter case models based on example data from able-bodied individuals could still serve as a seed for further optimization and fine-tuning of generated trajectories.

2.4.4 Real-time predictions

We performed a systems test with one subject (not included in the training, validation and test sets) to test the feasibility of using the Coordinated Movement (CM) controller in

real-time. The subject walked on a treadmill using the Open Source Leg (OSL) which was commanded using the reference trajectories generated by the CM controller.

The system showed low latency in predicting and generating the reference trajectories. The whole pipeline of operations resulted in a lag of less than 80ms in the response of the prosthetic leg. For upper-limb prostheses, a delay greater than 300ms is considered significant [30]. This value has not been established for lower-limb prostheses, but it can be generally deduced that lower delays are desirable, especially during load bearing phases. We expect the minimum inherent lag of this strategy to be lower than 80ms when optimized for a production-level implementation. Although training this kind of model can be quite computationally expensive, run-time implementations can be made to be significantly faster than what we report here.

In offline tests, normative biological limb trajectories were used as ground truth to evaluate prediction errors. For the real-time test, we don't have a ground truth trajectory. Fig. 2.15 depicts a representative 10-second window from the experiment. The predicted knee trajectory (red) appears appropriate, and the device can approximately track it (blue). However, we observed that the predicted trajectories were noisier than in offline experiments, characterized by greater variance and undesirable features such as the back-and-forth spikes at the extrema. These disturbances were more perceptible with both ankle and knee joints of the OSL being actively controlled, and sometimes compromised the stability of the participant. Substituting the active ankle joint with a passive ankle allowed the participant to walk more easily, although with occasional kickbacks.

To address these challenges, smoother command trajectories could be obtained to improve user's comfort and confidence in the system. Furthermore, we suspect that the noisier outputs could be due to noisier inputs from the motion capture system. Raw motion-tracking data is inherently noisy. A post-processing engine further improves data quality by including past, present, and future samples. While offline analysis has the benefit of using clean post-processed data, real-time control does not. This might result in poorer prediction performance.

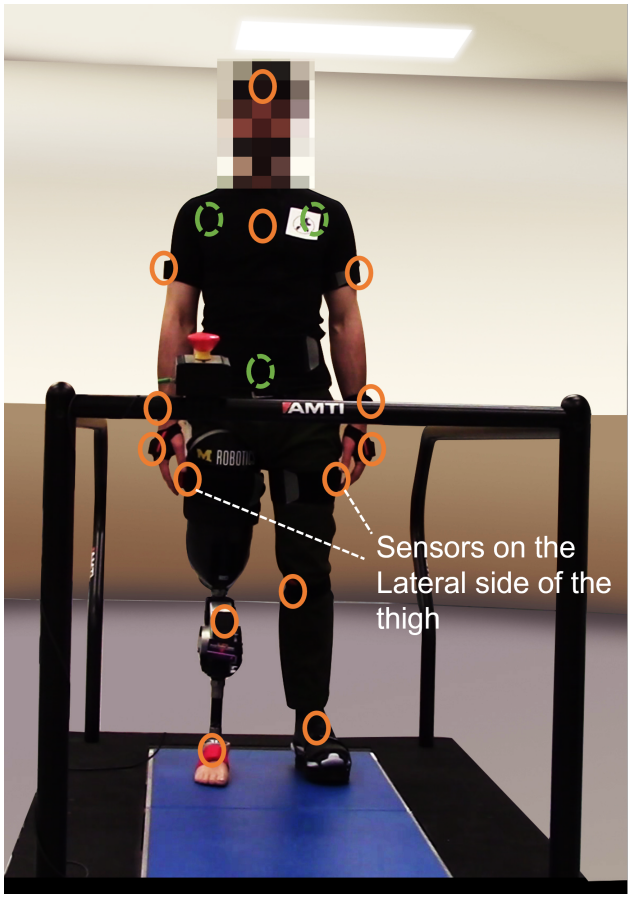

Figure 2.14: Real-time setup with Open Source Leg (OSL). The positions of imus are indicated using circles. Green dashed circles mean that sensors are placed on the posterior side of the body and are not shown in the figure. An individual wearing the motion capture sensors walked on the treadmill at a 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 knee-joint of the OSL in real-time. Ankle was locked.

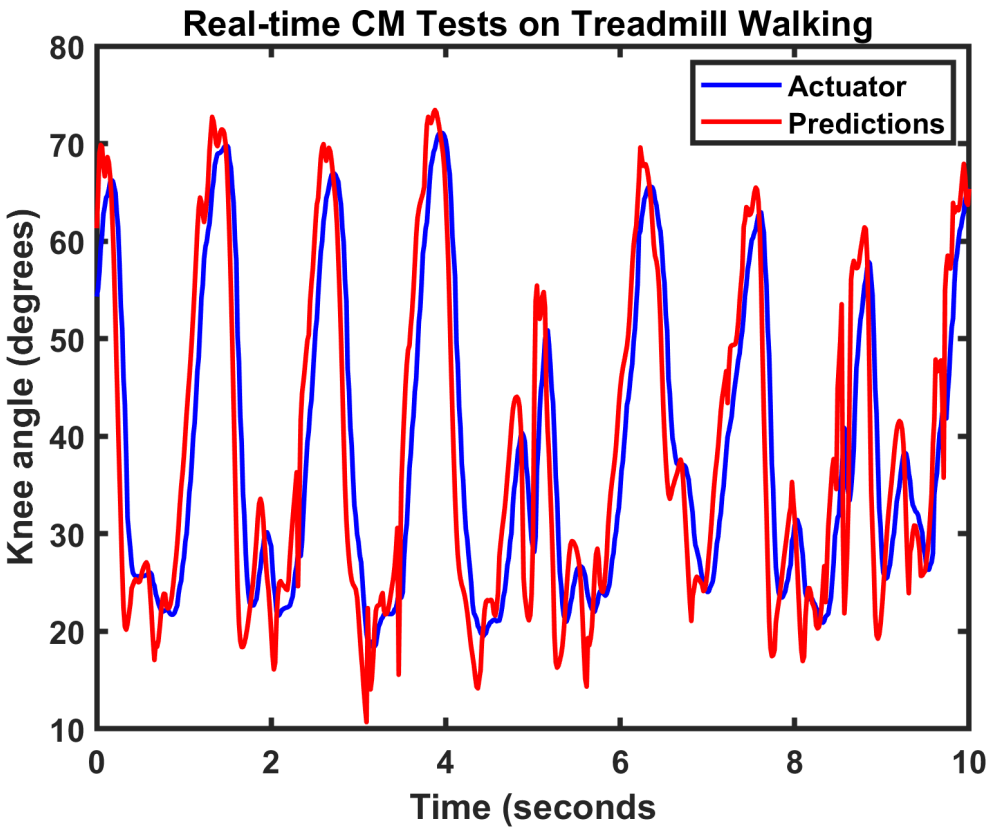

Figure 2.15: Knee joint predicted (red) and actuated trajectories (blue) during the real-time tests with treadmill walking activity. For this trial, the actuated trajectory had an RMS error of 8.1 degrees with respect to the network predicted trajectory (red).

Another challenge is that while a neural network like LSTM can be expressive, when it might fail and deliver an inappropriate prediction is hard to know. In the case of an assistive device this can be major safety issue. Thus, it is essential to be able to trust the prediction before it is executed on the hardware. Using uncertainty modeling [52] along with the predictions, might help determine whether to trust the network's predictions or to fall back on a safer strategy.

Chapter 3

ENVIRONMENT MODELING FOR GAIT PREDICTION USING EGOCENTRIC VISION

Abhishek Sharma, Eric Rombokas

Transactions on Neural Systems and Rehabilitation Engineering, 2022 [104]

Abstract

We seek to predict knee and ankle motion using wearable sensors. These predictions could serve as target trajectories for a lower limb prosthesis. In this manuscript, we investigate the use of egocentric vision for improving performance over kinematic wearable motion capture. We present an out-of-the-lab dataset of 23 healthy subjects navigating public classrooms, a large atrium, and stairs for a total of almost 12 hours of recording. The prediction task is difficult because the movements include avoiding obstacles, other people, idiosyncratic movements such as traversing doors, and individual choices in selecting the future path. We demonstrate that using vision improves the quality of the predicted knee and ankle trajectories, especially in congested spaces and when the visual environment provides information that does not appear simply in the movements of the body. Overall, including vision results in 7.9% and 7.0% improvement in root mean squared error of knee and ankle angle predictions respectively. The improvement in Pearson Correlation Coefficient for knee and ankle predictions is 1.5% and 12.3% respectively. We discuss particular moments where vision greatly improved, or failed to improve, the prediction performance. We also find that the benefits of vision can be enhanced with more data. Lastly, we discuss challenges of continuous estimation of gait in natural, out-of-the-lab datasets.

3.1 Introduction

In this chapter, we examine how the environment affects the prediction performance of the gait prediction model we presented in the previous section. We use the LSTM based predictive model and train it on a new full-body kinematics dataset captured in more challenging-uncontrolled and changing natural environments. We compare the prediction performance of the model in congested and free spaces. Then, we show how including an environment model using egocentric vision affects prediction performance. We find that vision (optical flow) can improve the prediction performance for both knee and ankle, with greater improvement in congested spaces.

Next, we discuss the motivation for this work. In sections 3.2 and 3.3, the methods are presented. Section 3.4 presents the key results, and finally the limitations and challenges are discussed in 3.5.

3.1.1 Motivation

Marker-based Motion Capture systems are the gold standard for gait data collection. They yield very accurate measurement of gait but require external cameras or emitters and an infrastructure for recording movement. Thus, it is impossible to record gait in real world environments using marker-based systems. As a result, most gait analyses have been limited to laboratories where it is difficult to simulate real world environments. Wearable motion capture based on inertial measurement units (IMU) offer a solution. Previously, the inherent drift associated with IMU sensing, prevented their use for gait analysis but in recent year the technology has matured [99] and as a result these systems are increasingly being used for gait analysis [12, 68].

We seek to use the progress in wearable motion capture, to understand how natural environments might affect the predictability of gait. Real world environments cannot be controlled and as a result humans have to adapt their gait to be navigate through them. Thus, a lot of variations that might arise in everyday ambulation can be captured, which was

not possible previously. These variations might also make gait prediction more challenging. Gait prediction system is an important component of assistive devices. Thus, it is important to understand how the predictive models of gait perform in the real world.

In addition to the wearable motion capture, wearable cameras and eye-tracking have also been increasingly used over the past decade, to study the human behavior. These devices enable us to study what is happening in the surrounding environment and how it is affecting a person's gait [68]. In this work, we seek to use wearable camera to augment our LSTM-based gait prediction model with an environment model. This shows **the potential of including vision in the continuous-control [82] of assistive devices. This study also serves as a proof of concept that computer vision techniques can be extended towards reasoning about human gait.** Next, we briefly review existing vision methods for assistive devices and provide context for our work. Then, in **Appearance and motion in computer vision**, we motivate the use of optical flow as the visual input to our model.

3.1.2 Previous work on computer vision for assitive devices

Previously, computer vision has been used in the high-level controller (See Fig. 2.1 for user-independent sensing of environment [131, 119, 118]. Combining environment information along with user-dependent neuro-mechanical sensors, the performance of locomotion mode recognition can be improved [121, 62, 56, 57]. Typically, environment recognition for prosthesis control has involved separate feature extraction and feature fusion [131]. Features from sub-regions of a depth image or point cloud are extracted and then an environment classifier is applied [67, 130, 128]. Convolutional layers have also been used to extract deeper features for robust performance in relatively complex environments [81, 132]. Recently, there has been greater interest in using deep learning methods for terrain and environment classification [59, 133, 58, 60]. Then, environment features or classification decision can be fused with activity modes decision, using decision trees or SVM [62, 67]. For a more exhaustive review of environment sensing techniques for terrain classification, refer to [131, 118, 60].

Environment sensing in pursuit of continuous control, however, has been less examined.

Recently, it was shown that gait phase can be estimated from depth camera data for level ground walking [130]. However, to our knowledge it has not yet been shown how successfully joint angles can be directly estimated in natural out-of-the-lab environments. We hypothesize that visual information from the environment can be predictive of future joint movements. Here, we aim to show that computer vision can help improve continuous estimation of joint angles, in an end-to-end manner.

3.1.3 Appearance and motion in computer vision

Computer vision has been previously used for egocentric human activity recognition [33, 79] as well as to learn visual representations from animal behavior [26, 27]. These approaches have used vision in two complementary forms: appearance and motion [107, 41, 19, 64, 108, 122]. Appearance refers to the visual features of the environment, such as the shape and texture of the terrain, the presence of objects, and the context of the scene. For example, the appearance of a cluttered classroom is different from that of a grassy field, in terms of colors present, the shapes and presence of objects, etc. This information can be obtained from a single frame. Motion, on the other hand, integrates information across frames to capture the dynamic nature of the scene. Thus, it focuses on the wide-field movement of the background, the unified movement of objects, and other dynamic, time-varying properties of the visual stream. To the best of our knowledge, these approaches haven't been used in direct prediction of joint angles.

Since we are primarily interested in the movement of the human, we focus on motion rather than appearance. We anticipate that this will provide the most salient information for predicting joint angles as the person navigates. A common way to reason about motion is to use *optical flow*. There are a variety of methods for computing optical flow [101, 47], but all of them seek to quantify change over time in the visual stream. The most simple way of thinking of optical flow is the change at each pixel between consecutive video frames. When taken as a whole, optical flow calculations provide different geometric features like shape of objects in the visual field, their location, and their movement directions and speeds

relative to the camera. Optical flow has been used in engineering and biology for decades, but for the purpose of this paper, we use a deep learning based approach, intended to provide information about the movement of the environment, the objects in the environment, and the user themselves.

3.2 Methods

3.2.1 Participants

Ambulation data was collected from a total of 23 healthy participants (12 females, mean age 22.8) with no 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. Data can be made available, via a data use agreement, upon request to the authors.

3.2.2 Activities

The subjects walked at a self-selected pace in 3 types of environments (Fig. 3.1): 1) Classrooms with a dense obstacle arrangement, 2) Atrium with sparse obstacle arrangement, and 3) Two Stairwells with 33-step staircases each. The arrangement of obstacles was not controlled, and varied slightly from trial to trial, and reflected the natural variation of the day-to-day rearrangement of furniture. Each trial consisted of the subject walking in each environment (except for one subject, for whom there is no classroom data). The order of traversal of the environments was randomized. Transitions between environments were also captured in the data. These included opening doors, flat ground to staircase transitions and vice-versa. Walking speed and the chosen path through the environment were chosen as preferred by the subject. If the subject started performing repetitive movements, e.g. always turning left on encountering an obstacle or moving in a fixed loop several times, then the experimenter would suggest that they move to a different location.

Figure 3.1: **Environments**: Classroom, Atrium and Staircases. Obstacles and other people using the spaces were not controlled.

3.2.3 Instrumentation

Locomotion data was collected using the Xsens Awinda suit [3], consisting of 17 body-worn sensors placed at each segment of the limbs, as well as sternum, sacrum, shoulder scapula, and forehead (See the xsens user manual [5]). After a system specified calibration, the software provides joint kinematics in a 3D environment. 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 [126]. Data sampled at 60 Hz, from a total of 22 joints in 3 anatomical planes (sagittal, frontal, transverse) was captured for each trial.

Egocentric vision data was collected using an eye tracker from Pupil Labs [51]. The vision and gaze data was captured at 30 Hz. We ignore gaze data in this study. No equipment was set up in the open spaces.

3.2.4 Experiments

For each data collection session, the subject was briefed about the task to be performed and the equipment to be donned, and informed consent was obtained. Subjects first performed a

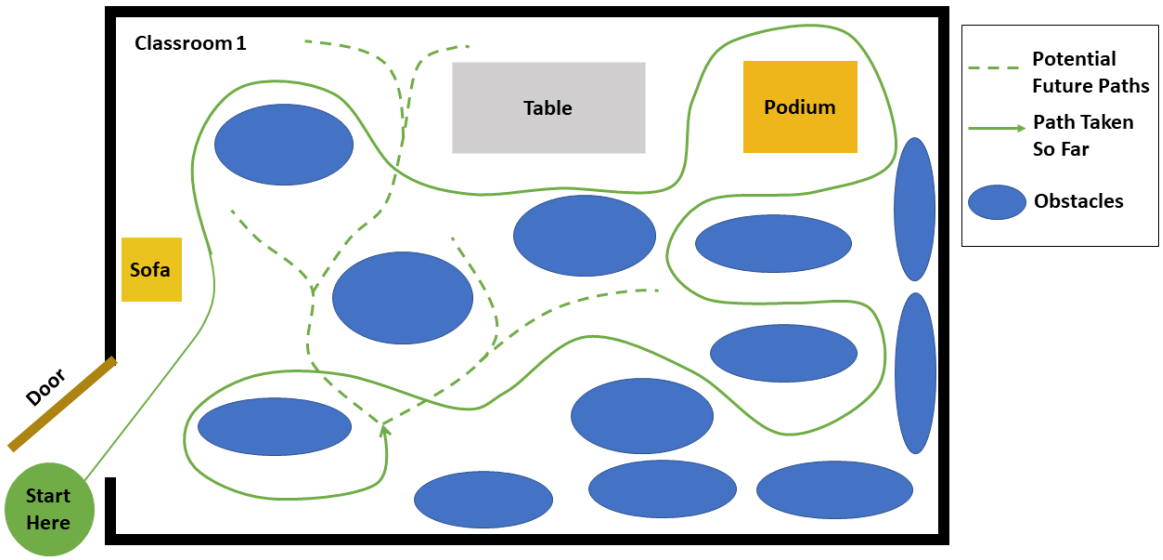

Figure 3.2: Classroom: Architecture of one of the classrooms. The arrangement of obstacles is not controlled and varies across subjects. The subject walks at self-selected speed and along self-selected path, The experimenter directed the subject to change their path only if the subject repeated the same path more than 2 times.

standard n-pose calibration routine as dictated by the device manufacturer. The subject also wore the pupil eye-tracker, and performed its calibration routine. Following this, the data collection began. The duration of each trial varies between 5-10 minutes. Participants were encouraged to take breaks after each trial. Duration of the entire setup and data collection was around 1.5 - 2 hours. On an average, 30 minutes of data was collected from each subject. **A total of 11.8 hours of walking data was collected.**

3.2.5 Data Processing

Xsens data was reprocessed using 'HD' processing feature, provided by the manufacturer for offline use, to enhance quality and remove noise.

Vision frames were extracted from Pupil eye-tracker video recordings using ffmpeg a free

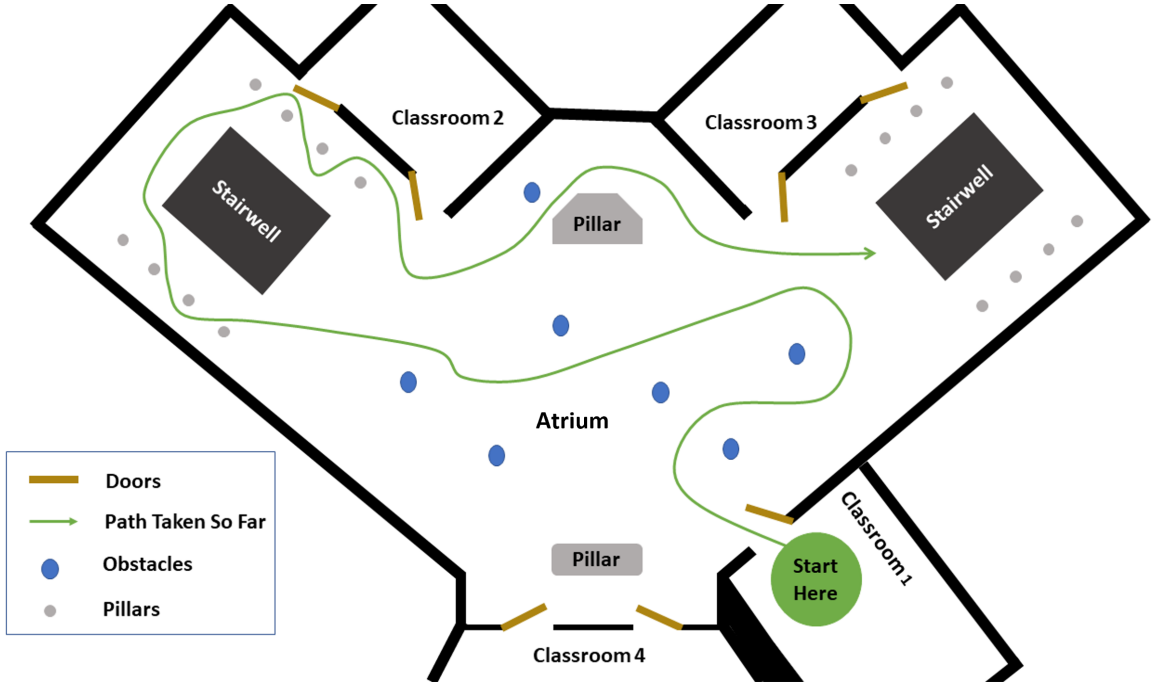

Figure 3.3: Atrium: Architecture of the Atrium. The arrangement of obstacles is not controlled and varies across subjects. The subject walks at self-selected speed and along self-selected path, The experimenter directed the subject to change their path only if the subject repeated the same path more than 2 times.

and open-source utility. The images were downsampled to a resolution of 320×240. The vision frames were synchronized with kinematics frames using UNIX timestamps. Since, vision frame rate was 30 Hz while kinematics frame rate was 60 Hz, we dropped kinematics frame which didn't have a corresponding vision frame. We visualized vision and kinematics data to validate the synchronization. Optical flow features were then extracted and stored as described in the following section.

3.2.6 Machine Learning Details

In previous work [83, 84], we used a Long Short Term Memory (LSTM) network to regress to joint angles for different activities like flatground walking, staircase and a standard high-mobility task called CHAMP. In this study, we use the same network architecture, shown in Fig. 3.4. However, in the present experiment we test the method on a more challenging out-of-the-lab dataset in natural environments. We then introduce vision as an augmentation to improve prediction performance.

Data from 22 subjects was split into training and validation sets. The trials from 3 randomly chosen subjects were used as the validation set. Data from one held out subject who was not included in the training and validation sets, was used as the test set. This choice was made because this work is oriented towards developing a prosthesis controller, which might be used by a previously unseen user and in an environment different from training set. Consider a time series trajectory of 57 intact joint angles $x \in \Re^{57\times10}$, where

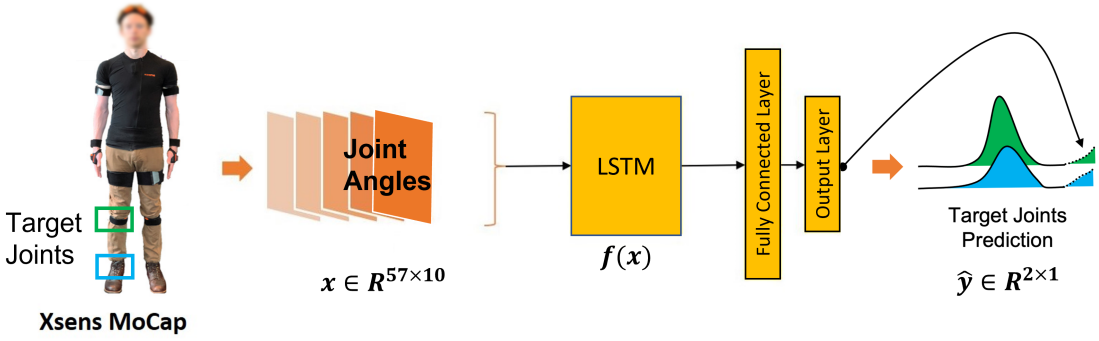

Figure 3.4: **Rai et al.**: Neural Network Architecture from the previous work on Coordinated Movement. The network is trained on our new dataset. The kinematics sequence is processed by a LSTM network followed by a fully connected layer to predict the knee and ankle joint angle

57 is the number of coronal, saggital, and transverse plane angles for 19 joints including left lower limb hip, knee, ankle, ballfoot, the right hip as well as upper body joints (See the xsens

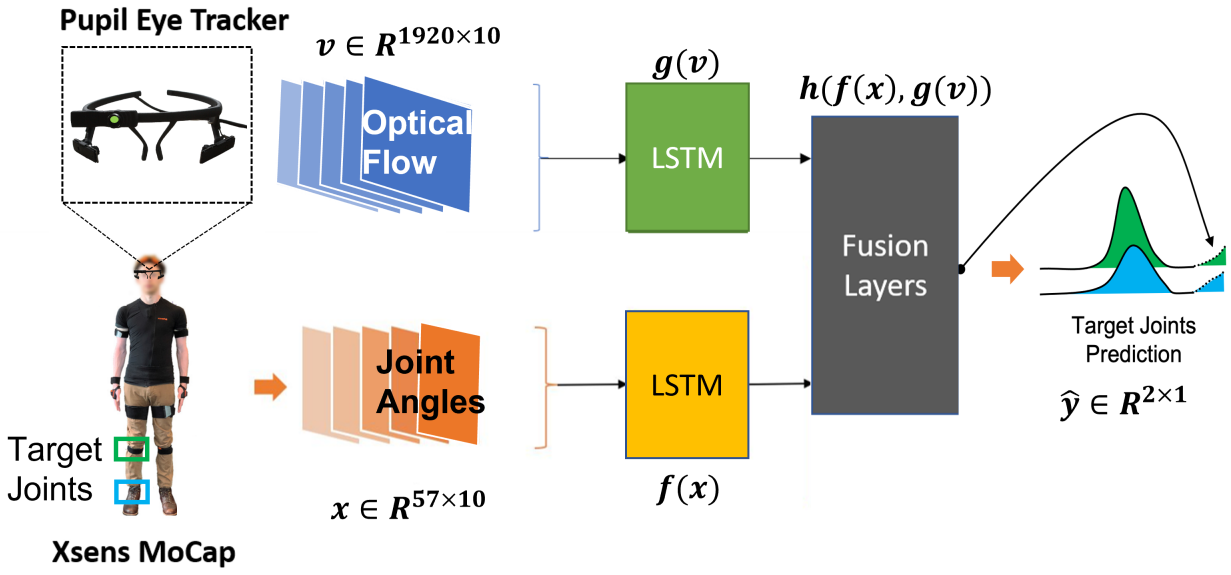

Figure 3.5: **Optical Flow Net**: Neural Network Architecture for fusion of optical flow and kinematics. Optical Flow features are extracte using the PWC-net [112], trained on the MPI-Sintel dataset [15] (See Optical Flow Feature Extraction). The flow features from all the frames are processed by a LSTM network. The outputs of the vision and kinematics LSTMs are combined using fusion layers to predict knee and ankle joint angles. For the *No-Flow Net* baseline, optical flow features are set to zero.

user manual [5] for more details), and 10 is the number of timesteps. The LSTM network model (Fig. 3.4) estimates the instantaneous target joint values $\hat{y} \in \mathfrak{R}^{2\times1}$ for the saggital plane angle of knee and ankle of the right limb. This target is chosen to be the angle *at 1 second* (approximately 1 gait cycle) into the future.

$$\hat{y} = f(x) \tag{3.1}$$

where f is the LSTM network.

For the vision model (Fig. 3.5), the optical flow features $v \in \mathfrak{R}^{1920\times10}$ are processed with a separate LSTM and the output of the two LSTMs are combined using a fusion module of fully connected layers. The number of hidden units in the successive layers of the fusion network is half the number in previous layers e.g. for an input of dimension N, the first layer of h will have N hidden units, second layer N/2, third layer N/4 and so on. Finally an output layer projects to the output dimension of 2.

$$\hat{y} = h(f(x), g(v)) \tag{3.2}$$

where f, g, h are the kinematics, vision and fusion respectively.

In other words, during every step, new joint angle data and new visual data flow into their respective recurrent LSTM networks (f and g in Fig. 3.5). The outputs of these networks combine with a fusion network (h) that learns to best combine their contributions for prediction. The temporal dynamics of each modality are accommodated in the rolling time window of inputs, their featurizations (especially for optical flow, in which featurization already tends to emphasize changes in time), in the recurrent dynamics of the networks, and the way the LSTM outputs are combined in the fusion network. For kinematics data we apply max-min normalization, which was also used in the previous work [83, 84]. This prevents high-ROM joints from dominating predictions, and scales all joint values between 0 to 1.

Optical Flow Feature Extraction A variety of methods have been developed to compute optical flow from videos [101, 47]. These can be used to generate high fidelity flow maps. However, due to computational and storage limitations with $\sim 10^6$ frames in our dataset, we decided to use an off-the-shelf pretrained optical flow prediction network, PWC-Net [112], to generate low resolution optical flow maps. We use the optical flow output from the 2^{nd} level (lower resolution) of the PWC-Net [112]. At this level of processing, the features are low-resolution optical flow maps that summarize the optical flow present in relatively large (compared to pixel-level) portions of the visual scene. The output features are of size $24 \times 40 \times 2$. In other words, the camera data at a resolution of 320×240 is input to the PWC-Net and converted into two optical flow channels of size 24×40.

We visually inspected predictions of the network pre-trained on CityScapes [21], Kitti12 [40], Kitti15 [69], and MPI-Sintel [15] datasets. We found the network trained on Sintel dataset to have best optical flow predictions on our dataset. This is expected, as the Sintel dataset has a greater variety of camera movements, pertaining to people and objects moving in a variety of indoor and outdoor environments, while the rest of the datasets are generated from cameras mounted on cars for autonomous driving applications. The lighting conditions on the Sintel dataset are also more diverse. When transferring a pre-trained network, the data should be normalized based on the statistics of the data the network was pre-trained with. Therefore, we normalized our vision frames using mean and standard deviation of MPI Sintel dataset rather than those of our dataset.

Rolling Time Window LSTMs are trained by error backpropagation through a desired number of time steps (sequence length). Selecting an optimal sequence length is required for optimal performance as shorter sequences could miss out on crucial information to predict the output, while longer sequences are computationally expensive. We generate input samples of sequence length 10, using an overlapping rolling window on the time series data. A similar choice exists for how far into the future to predict. We chose 1 second (approximately one gait cycle) for simplicity. We do not present an exhaustive optimization for these factors

here, but generally found the results were insensitive to sequence lengths between 2 and 30. Similarly, we have found that performance decreases the further into the future one attempts to predict. The contribution of this chapter is to showcase the relative improvement that can be obtained by including visual data, but not to demonstrate absolutely optimal performance.

Loss Function and Neural Network Hyperparameter Optimization Mean squared error (MSE) between the predicted and measured knee and ankle joint angle was used as the loss function to be optimized. The hyperparameters from Rai et. al were used for the size of the kinematics LSTM (f) , and were not optimized anew. This is for maximum comparability with Rai et al., and because they were optimized for that previous work [84]. Random search was applied to optimize the other hyperparameters. We optimized over the following: the number of units in each of the two layers of the vision LSTM, the regularization parameter for L2 loss, and learning rate. The performance of the model was evaluated on the validation set after each epoch. The training was stopped if performance did not improve for 20 consecutive epochs. The best performing model on the validation set was saved and used to generate predictions and metrics on a test set. The range of parameter values tested is shown in Table 3.1. The optimal hyperparameter sets for each network were used to compare and evaluate performance.

3.3 Analysis

After all data was collected and processed, predictions were generated offline. We used normalized Root Mean Squared Error (RMSE) and the Pearson correlation coefficient (PCC) as outcome measures. These were calculated between the actual measured joint angles and the predictions of each of the regression strategies at 1 second ahead in time. Measures are reported for different activities and quantity of data. Ideally, the RMSE would be 0 and PCC would be 1.

Table 3.1: Hyperparameter values tested for Vision LSTM (g) and Fusion network (h). The kinematics LSTM (f) has 2 layers and 32 hidden units, following the hyperparameter tuning in [84].

Hyperparameter	**Range/Values**	**Optimal**
Learning Rate	$[10^{-6} : 10^{-1}]$	10^{-5}
Batch Size	[100]	100
Number of Epochs	Max 1000 with early stopping	-
Number of LSTM (g) layers	[2]	2
Number of Hidden Units in g	[32,64,128,256,512]	64
Number of fusion (h) layers	[2,4]	4
Regularization Rate	[0,0.001,0.01,0.1]	0

3.3.1 Baselines for improvement

To determine how much the Optical Flow model improves performance, we compared against two baselines:

- **Rai et al:** This is the model (see Fig. 3.4) used in our previous work for structured activities. We establish performance of this model on our natural out-of-the-lab dataset. We show that using optical flow improves performance over this baseline.

- **No-Flow model:** To rule out improvements due to difference in the architecture and test the contribution of optical flow only, we perform ablation by replacing optical flow vectors with zeros of same size. This prevents any information flow from vision to the fusion layer [70].

Our models were trained with all the training data without splitting into separate activities. This is because our goal is to design a single unified continuous controller. To report performance on individual activities, we separated all activities manually after train-

ing. This is because in a dataset with a diverse set of activities, average performance can obscure activity-specific performance.

Significance testing

MATLAB was used for significance testing. To test the significance of our results, we split the entire test dataset into non-overlapping sections of length 7 seconds (at least 5 gait cycles). Then, we generated predictions from the three models and computed RMSEs on each sample. First, we tested for the Gaussianity of the error distributions, using the Kolmogorov-Smirnoff test [115]. They were found to be non-Gaussian, so a paired sample T-test was not applicable. Therefore, we use Wilcoxon's Signed Rank Test [116] on paired samples of errors to test significance.

3.3.2 Dependence on data

A major challenge with using data-driven machine learning methods, like the ones we present here, is knowing whether the dataset is sufficient for generalization, and whether the performance is representative of the ultimate limits of the method. In order to show how sensitive the results we present here are to the amount of data, we designed some analyses in which we vary the amount of data available. We assessed performance for different amounts of training data by varying:

- The number of subjects included in training data.
- The percentage of data included from *each* subject in training data.

The first test helps us understand how adding additional unique subjects could affect performance, and the second helps us understand whether longer sessions, or perhaps additional sessions, would affect performance. We do not present additional analyses here, but in principle one could assess sensitivity to the amount of data from particular activities or environments, types of movements, etc.

3.4 Results and discussion

In this section, we present key results towards understanding the role of environment in gait prediction and how utility including environment model in prediction. Tables 3.2-3.5 show the prediction performance for the networks: The two kinematics networks for baseline **Rai et al.** and **No Flow**, and the network that fuses Optical Flow with kinematics, named here as **Optical Flow**.

3.4.1 Predicting gait in congested environments is harder

From Tables 3.2-3.5, we see that RMSE is lower for atrium compared to the classroom for both the kinematics networks: **Rai et al.** and **No Flow**. This could be because the classrooms (congested space) require more drastic and abrupt changes in the walking direction due to narrower space available for movement. Since we are predicting at 1 second in the future, the information about an abrupt turn would be absent in the history of body movement. Atrium has freer space and the participant can take smoother turns. The PCC is also smaller for predictions in the Classroom.

3.4.2 Fusing vision and kinematics improves continuous prediction of joint angles

RMSE and PCC performance are statistically significantly better using the Optical Flow network for classroom, atrium, and overall, but not for stairs. Tables 3.2 through 3.5 show performance for knee and ankle RMSE and PCC between the measured joint angles and the predictions provided by each of the 3 regression methods. The method with best performance for each row is indicated in bold font. Asterisks indicate the difference between optical flow and no-flow networks is statistically significant ($p < 0.05$) using Wilcoxon's Signed Rank Test. **The improvement is most prominent for classroom walking, where obstacles are more densely packed and the environment greatly constrains how the participants moved.** Even though the ankle RMSE is lower than Knee RMSE (Tables 3.2 and 3.3), the ankle predictions are less correlated with the actual measured trajectories

Table 3.2: Knee Predictions: Root Mean Squared Error Between the Measured Joint Angles and each of the Three Predictions. The method with best performance for each row is indicated in bold font. Asterisks indicate the difference between optical flow and no-flow networks is statistically significant ($p < 0.05$) using Wilcoxon's Signed Rank Test.

	Rai et al.	**No-Flow**	**Optical Flow**
Classroom	0.129 ± 0.035	0.135 ± 0.043	**0.109 ± 0.039 ***
Atrium	0.119 ± 0.025	0.117 ± 0.027	**0.109 ± 0.035 ***
Stair Up	**0.187 ± 0.054**	0.198 ± 0.046	0.194 ± 0.055
Stair Down	0.238 ± 0.020	**0.224 ± 0.021**	0.248 ± 0.026
Overall	0.140 ± 0.047	0.140 ± 0.049	**0.129 ± 0.057 ***

Table 3.3: Ankle Predictions: Root Mean Squared Error Between the Measured Joint Angles and each of the Three Predictions. The method with best performance for each row is indicated in bold font. Asterisks indicate the difference between optical flow and no-flow networks is statistically significant ($p < 0.05$) using Wilcoxon's Signed Rank Test.

	Rai et al.	**No-Flow**	**Optical Flow**
Classroom	0.112 ± 0.026	0.114 ± 0.030	**0.098 ± 0.028 ***
Atrium	0.111 ± 0.023	0.104 ± 0.033	**0.093 ± 0.028***
Stair Up	**0.144 ± 0.037**	0.148 ± 0.051	0.165 ± 0.038
Stair Down	0.240 ± 0.043	0.259 ± 0.052	**0.238 ± 0.037**
Overall	0.128 ± 0.047	0.128 ± 0.055	**0.119 ± 0.052***

(Tables 3.4 and 3.5). We again observe predictions to be better for atrium (freer space) when compared to classrooms (congested space).

Table 3.4: Knee Predictions: Pearson Correlation Coefficients Between the Measured Joint Angles and each of the Three Predictions. The method with best performance for each row is indicated in bold font. Asterisks indicate the difference between optical flow and no-flow networks is statistically significant ($p < 0.05$) using Wilcoxon's Signed Rank Test.

	Rai et al.	**No-Flow**	**Optical Flow**
Classroom	0.822 ± 0.087	0.841 ± 0.101	**0.873 ± 0.101***
Atrium	0.862 ± 0.063	0.898 ± 0.057	**0.907 ± 0.056***
Stair Up	**0.662 ± 0.238**	0.642 ± 0.219	0.647 ± 0.248
Stair Down	0.464 ± 0.126	**0.503 ± 0.160**	0.252 ± 0.175
Overall	0.773 ± 0.155	0.787 ± 0.187	**0.799 ± 0.205***

Table 3.5: Ankle Predictions: Pearson Correlation Coefficients Between the Measured Joint Angles and each of the Three Predictions. The method with best performance for each row is indicated in bold font. Asterisks indicate the difference between optical flow and no-flow networks is statistically significant ($p < 0.05$) using Wilcoxon's Signed Rank Test.

	Rai et al.	**No-Flow**	**Optical Flow**
Classroom	0.606 ± 0.212	0.646 ± 0.187	**0.705 ± 0.166***
Atrium	0.661 ± 0.095	0.749 ± 0.089	**0.777 ± 0.078***
Stair Up	0.584 ± 0.191	**0.603 ± 0.169**	0.364 ± 0.236
Stair Down	**0.063 ± 0.391**	-0.116 ± 0.476	0.059 ± 0.358
Overall	0.567 ± 0.255	0.567 ± 0.264	**0.637 ± 0.264***

3.4.3 Vision: Successes and Failures

While average RMSE reported in Tables 3.2 and 3.3 show modest overall improvements of 7.9% and 7.0% for knee and ankle predictions using optical flow, the average performance can hide interesting moments where vision drastically improved performance or failed to

improve performance. The data itself is heterogeneous and has changing environments. Thus, we report interesting moments where vision led to improved performance, and also where it did not. While differences in performance between optical flow and no flow for knee and ankle are similar as seen from table 3.2-3.5, here we will show only knee. This is because the differences are qualitatively easier to see for the knee trajectories since the correlations are higher. In Figs. 3.6-3.12, we present the Actual Trajectory (the measured kinematics), the predictions of the No Flow network (baseline network in which optical flow inputs are set to zero), and Optical Flow (the full network as shown in Fig. 3.5.)

Cases where vision improved performance: Fig. 3.6-3.9 show moments where vision captured critical maneuvers. These included approaching and opening doors, turning, avoiding obstacles and walking through narrow spaces.

Cases where vision did not help Failures are shown in Fig. 3.10-3.12. In Fig. 3.10, the participant suddenly changed plans and took a sharp left turn, approaching a classroom door. The illumination near the door was very poor until the door was opened and the participant entered the classroom. In Fig. 3.11, the participant approached a classroom door but instead of entering, stopped and turned in place to look at the experimenter before turning back again and opening the door. The performance during the turn is worse for the optical flow network. This could be because this maneuver is an outlier, where the visual behavior was not very predictive of body movement, and the optical flow confounded the predictor. The last example is shown in Fig. 3.12, where the participant descends stairs. The performance of both networks is poor. However, the RMSE value for optical flow network is worse.

3.4.4 Adding more data can further amplify the benefits of using vision

Fig. 3.13 shows performance of the No Flow and Optical Flow networks as a function of data size. the optical flow network outperforms the no flow network by an increasing margin as the

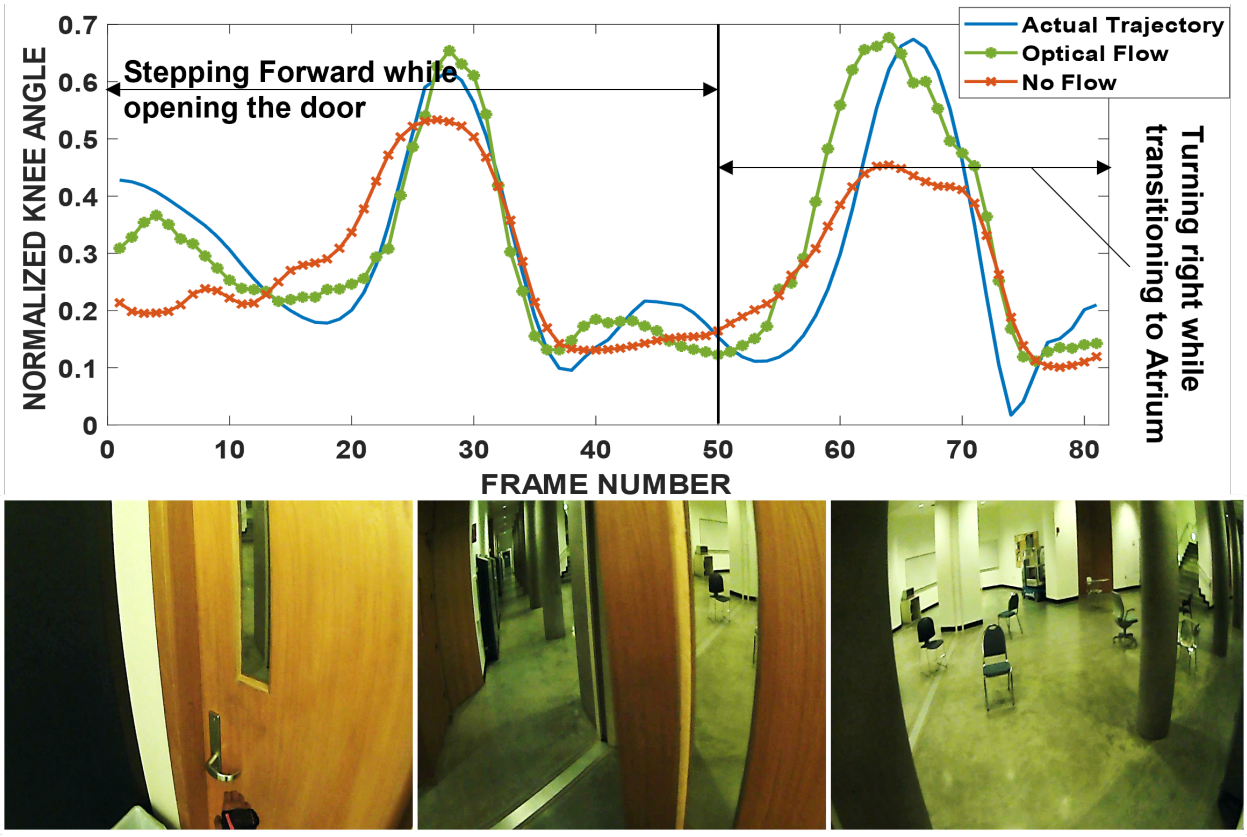

Figure 3.6: **Success:** Gait kinematics (above) and the first, 40th, and 80th vision frames (below) during the maneuver. The subject was exiting the classroom and entering the atrium. Optical Flow RMSE = 0.082, No Flow RMSE = 0.110.

amount of data increases. Thus the benefits of including visual data become more prominent as the data size is increased. Since this trend has not leveled out for the data considered here, we would expect improved performance for even larger datasets. There is reason to believe that egocentric video will soon become dramatically easier to collect, and will be available on an unprecedented scale [58, 18], and interest in wearable devices for consumer applications continues, e.g. [6]. Based on our analysis of the sensitivity to the amount of data (Fig. 3.13) we anticipate that performance improvements will continue with larger datasets. This is critical for moving this technology forward, because the experiment we present here took

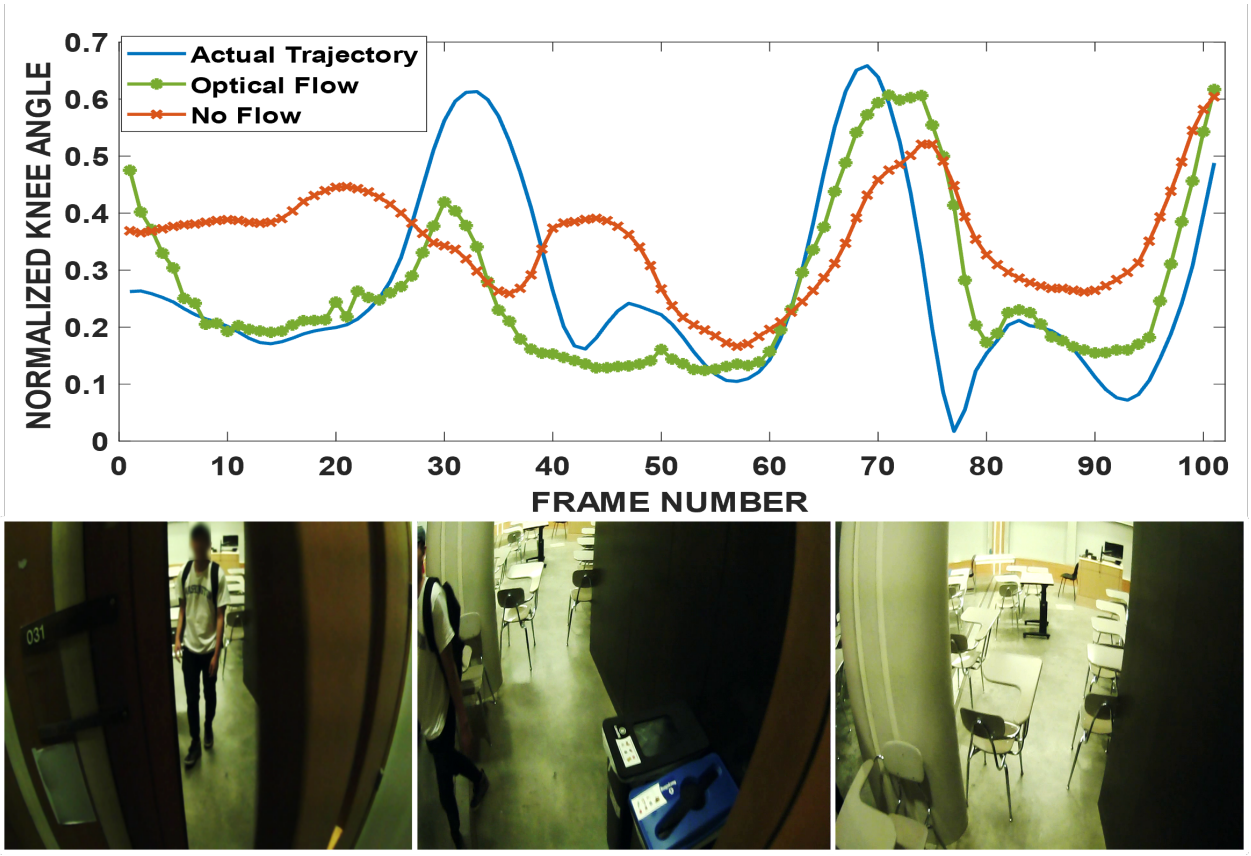

Figure 3.7: **Success:** Frames number 15, 60 and 100 are shown. The subject opened the door and entered the classroom. Optical Flow RMSE = 0.131, No Flow RMSE = 0.188.

place in only three different environments. The possible visual appearance and variability in lighting changes, materials, etc, will require more data that is representative of a wider variety of situations.

3.5 Limitations and challenges

3.5.1 Using generated trajectories as reference for control of assistive devices

The feasibility of translating continuous joint angle predictions into an appropriate control signal for a prosthetic limb is yet to be established, particularly for non-rhythmic activities.

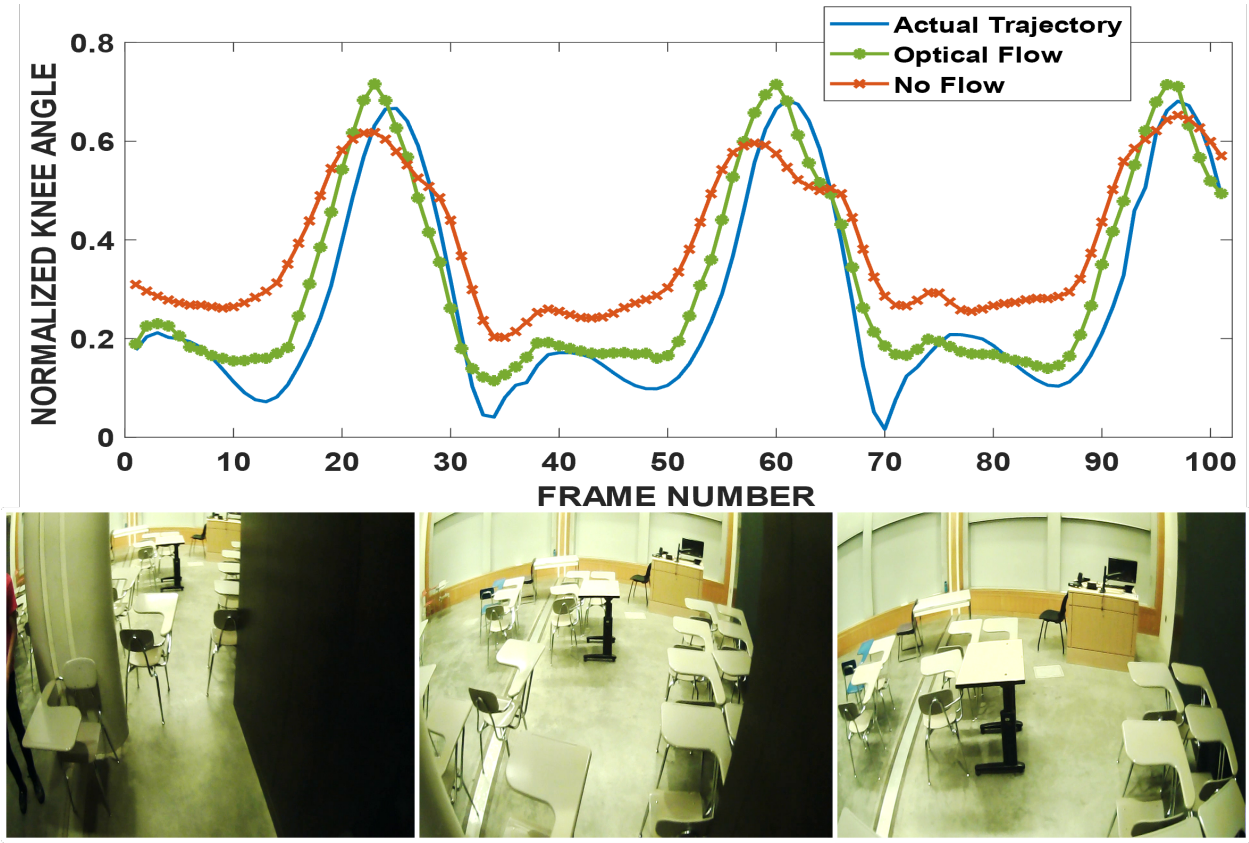

Figure 3.8: **Success:** In the follow up to the previous example, the subject moved forward in the classroom, through a narrow space between chairs. Frame number 1, 50 and 100 are shown. Optical Flow RMSE = 0.078, No Flow RMSE = 0.151.

Here we have focused on estimating intact participants' joint angles of the intended prosthetic joints, but using those estimations as reference trajectories is a distinct, and non-trivial engineering task.

Even though this dataset is relatively large, both in terms of number of subjects and data per subject, the prediction performance is probably not adequate for use in a prosthetic controller (especially for the ankle). It remains to be seen what the standard is for "close enough" for continuous joint angle regression.

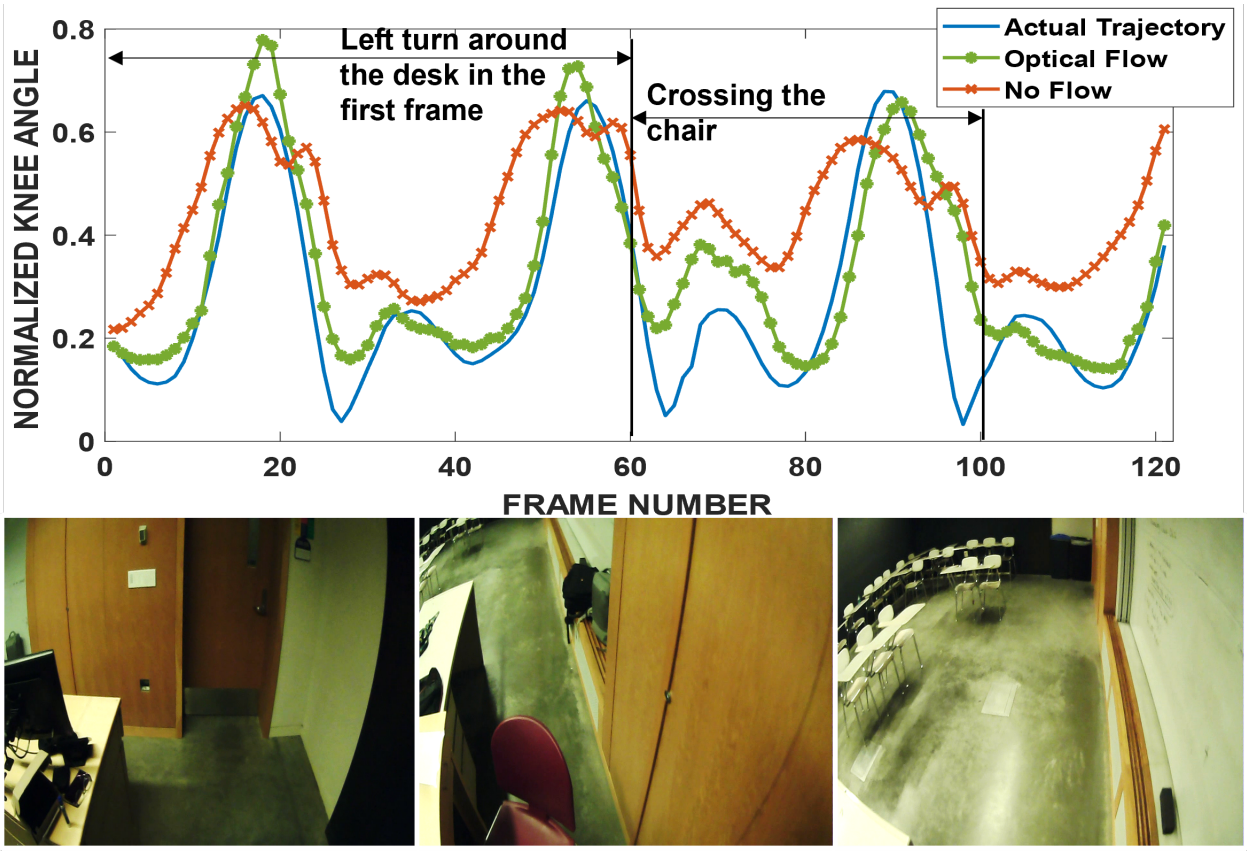

Figure 3.9: **Success:** 1st, 60th and 120th frames are shown in the figure. The subject executed a left turn around the desk shown in the first frame. The performance of vision is better throughout the maneuver. Slight decrease in vision performance occurs while crossing the chair shown in 45th frame, but is still better than no flow. Optical Flow RMSE = 0.097, No Flow RMSE = 0.204.

3.5.2 Data imbalance

Another limitation of this study is the data imbalance between different environments. Relatively lower amounts of data on stairs yielded inconclusive results on stair ascent and descent, as predictions for both vision and no vision networks are bad. Similarly, transitions between terrains, idiosyncratic movements and less frequent events such as opening doors, sit-to-

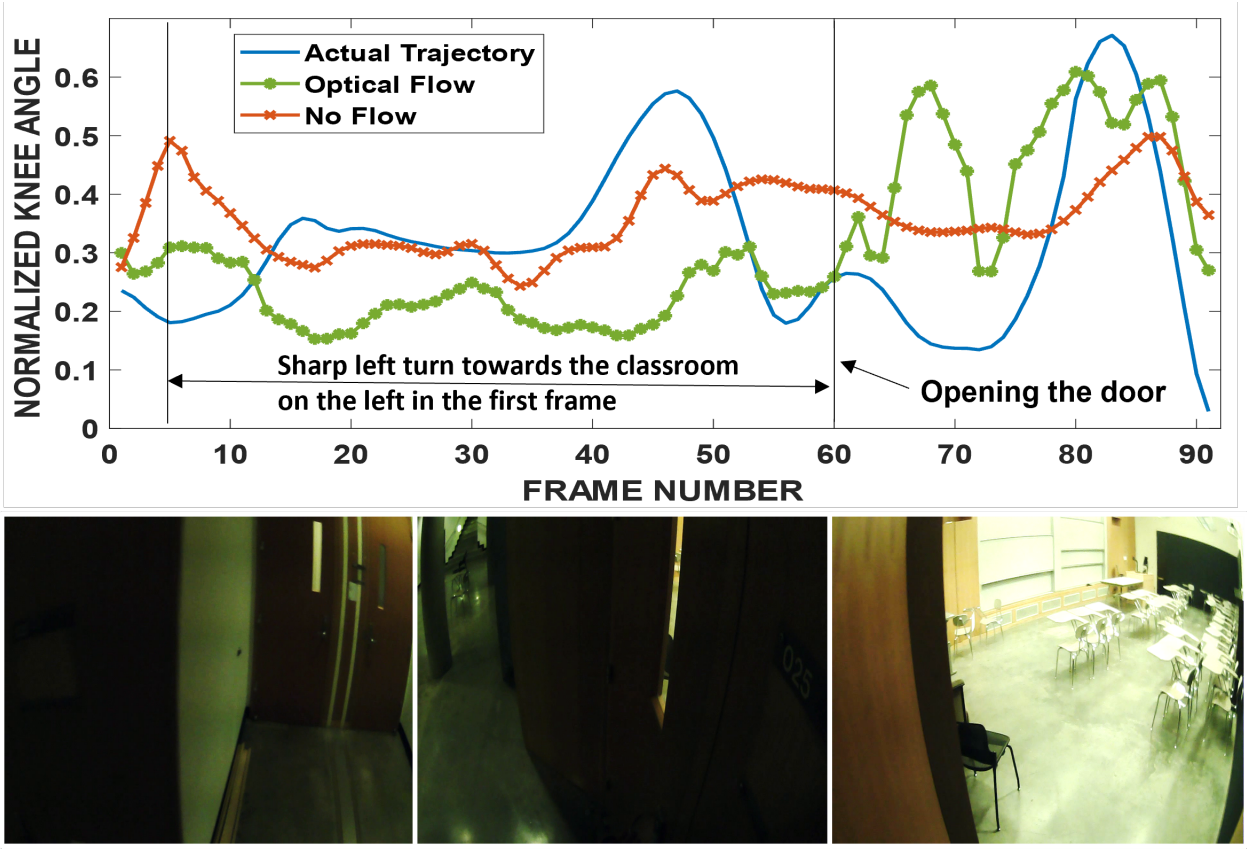

Figure 3.10: **Failure:** 1st, 45th and 90th frames are shown. This example shows an irregular maneuver where vision did not help. The subject was approaching the door on the right side of the 1st frame, but took a sharp left turn and instead approached the classroom on the left. The illumination was bad during this approach, and performance is worse using vision. Optical Flow RMSE = 0.187, No Flow RMSE = 0.148.

stand, etc, are present in the dataset. Performance during such events is reduced because there is less data to train for those unique moments. This is a well-established (see e.g. [55]) problem in machine learning, and there are some techniques that can be used to ameliorate the effects of imbalanced datasets, which we expect would improve results.

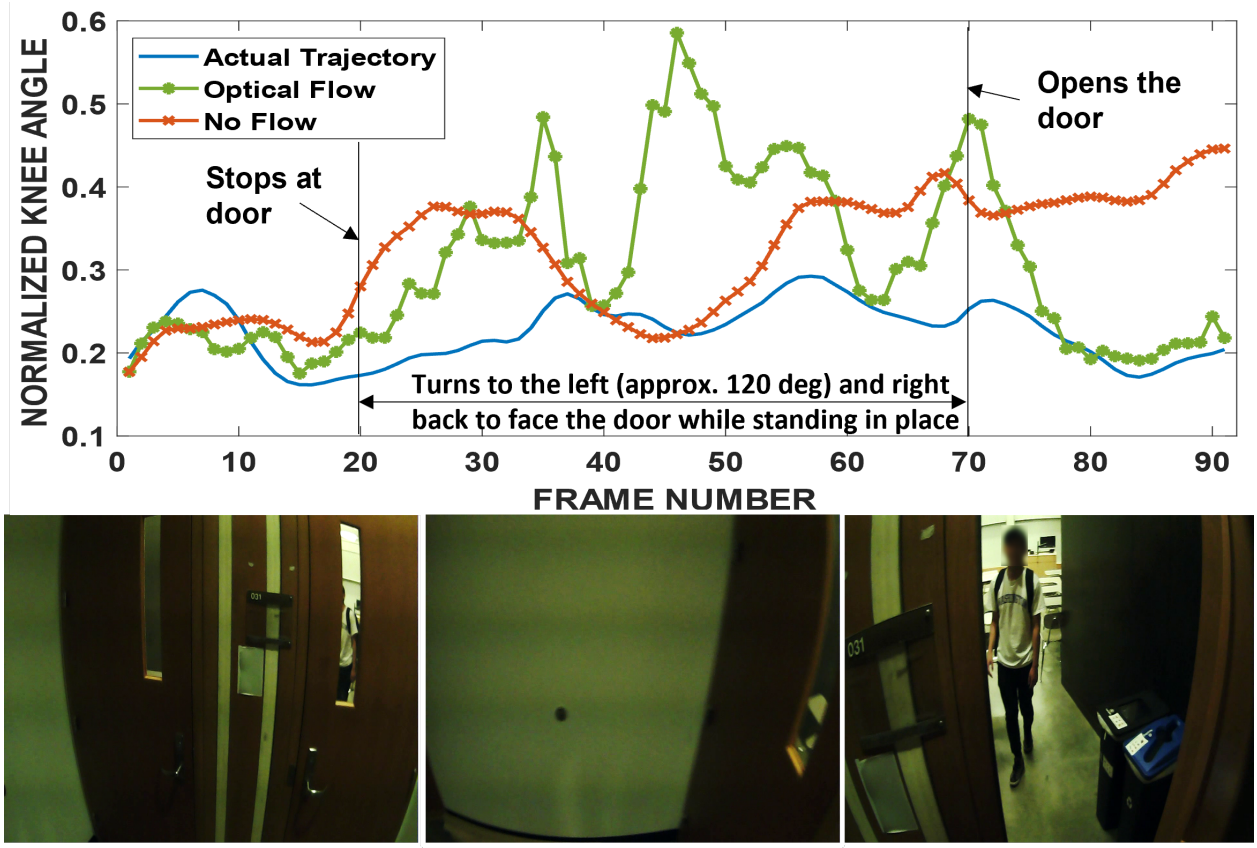

Figure 3.11: **Failure:** 1st, 45th and 90th vision frames are shown.The maneuvre is depicted in annotations. Optical Flow RMSE = 0.134, No Flow RMSE = 0.098 for predictions from frame 0 to frame 70 i.e. when the subject started opening the door.

3.5.3 Vision features are not optimal

These results are not necessarily intended to demonstrate the performance limits of this approach, but to demonstrate the possibilities that could come from adding egocentric camera data to body motion capture. We used an off-the-shelf optical flow generation process, not tailored to this particular application, but it would be straight foward enough to optimize that process. There also could be variations in camera orientation, resolution, etc. that could be leveraged to improve the utility of egocentric vision. Nevertheless, this experiment

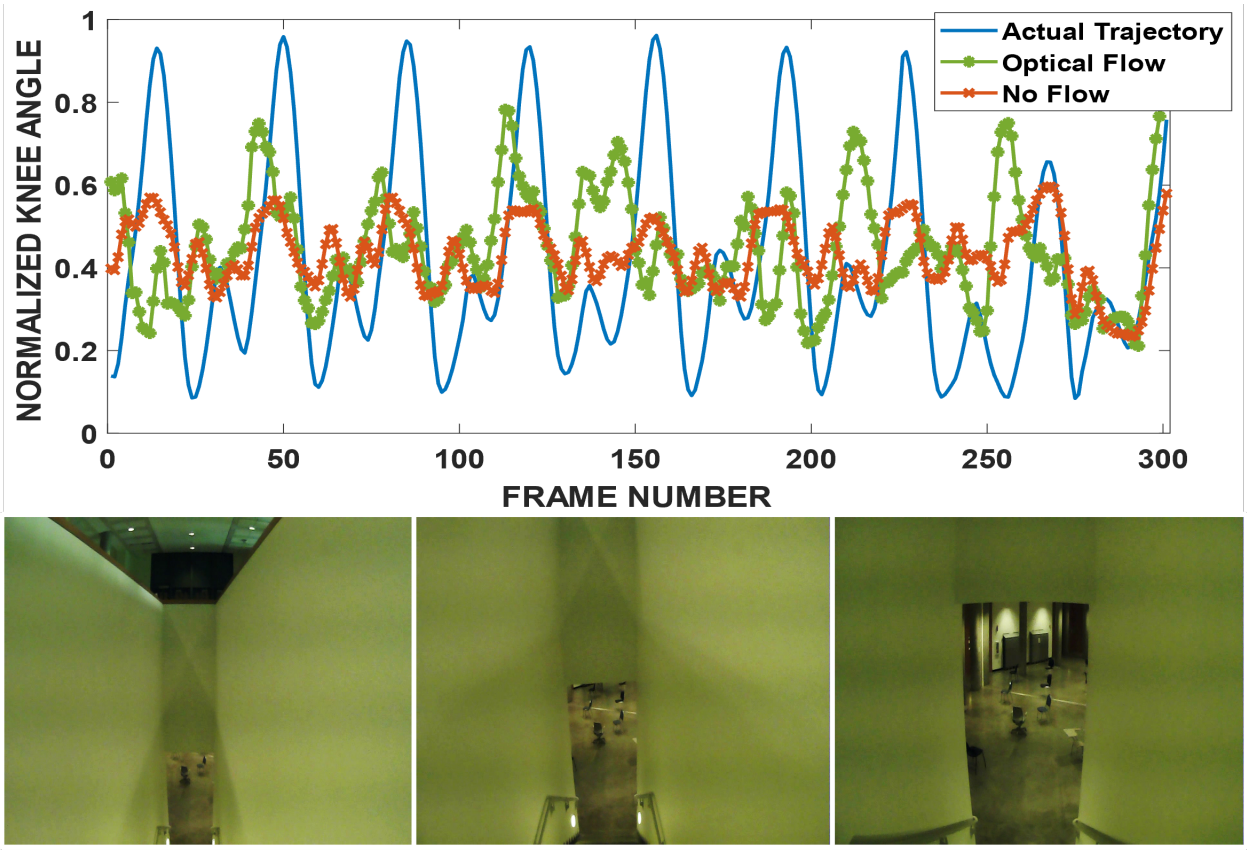

Figure 3.12: **Failure:** First and last vision frames are shown. The subject was descending staircases in this example. The performance of both networks is bad. Optical Flow RMSE = 0.269, No Flow RMSE = 0.211.

demonstrates the potential for improvement is there.

3.5.4 Prediction gait for a completely new subject is challenging

The results we present here are for performance of the predictors on a subject that was completely held out of the training and validation set. They are representative of how this approach would perform on an unknown user with no prior training examples. This is a challenging version of the application, because the system must generalize to user-specific gait style, behaviors like head movements and other tics, as well as to an uncontrolled changing

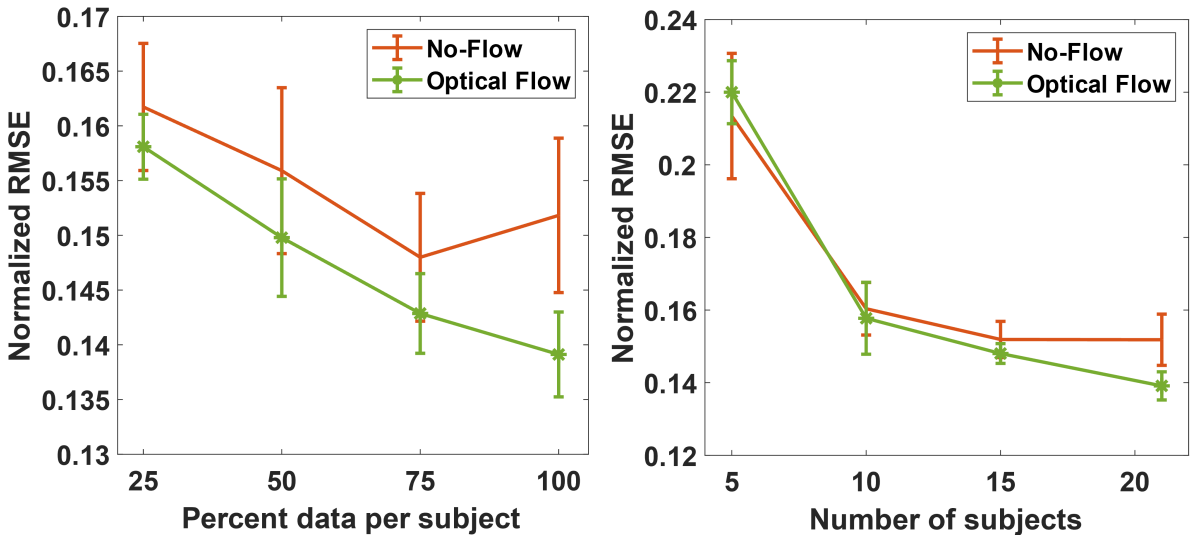

Figure 3.13: (left) RMSE as a function of percent data used per subject. With more data, the optical flow network improves more than no flow model. (right) RMSE vs. number of subjects in the dataset. As the number of subjects increases, the optical flow network performs increasingly better than the no flow network.

environment. Tuning the system to a new user using an approach similar to that highlighted in [124] or by including some brief walking data from the user in the training set, might help performance. In addition, the body dimensions of the user, (especially height) also influence where the eye-tracker is placed. Thus, fusing body dimensions or other measurements with the visual stream might further help performance, as in [34].

3.5.5 Performance on staircases

Vision did not improve prediction performance on staircases. This could be because vision seems to provide little additional information as subjects tend to move straight down the stairs without the broad choices of path selection they are presented with for the other environments. Obstacles and people on the staircases, where vision could prove most valuable, are relatively rare in the dataset and thus minimally impact RMSE. There could also be

other factors related to head orientation and gaze when navigating stairs.

3.5.6 Appearance and Motion

As discussed in the motivation, there are two general approaches to computer vision - appearance and motion. Here we have emphasized motion with the use of optical flow. There is evidence that humans use optical flow for steering towards the intended goal [123]. Here, we showed that it can help predict joint angles. However, inclusion of appearance could have further advantages and interesting capabilities.

3.5.7 Indoor environments

With this study, we first needed to establish that vision information can in fact be predictive of joint kinematics. Vision data is very high dimensional with several factors of variations like brightness, contrast etc. in addition to factors that actually affect movement like obstacles, people and terrains. To account for these factors we might have needed orders of magnitude more data than we have the capacity to collect currently. Thus, we limited our analysis to indoor environments where lighting is relatively stable. With datasets like Ego4d [6] (3000 hours of egocentric vision data), it might be possible to overcome this limitation.

Chapter 4

COMPLEXITY ANALYSIS OF LOCOMOTION ACTIVITIES

Abhishek Sharma, Eric Rombokas

Frontiers in Bioengineering and Biotechnology, 2022 [103]

Abstract

Gait complexity is widely used to understand risk factors for injury, rehabilitation, the performance of assistive devices, and other matters of clinical interest. We analyze the complexity of out-of-the-lab locomotion activities via measures that have previously been used in gait analysis literature, as well as measures from other domains of data analysis. We categorize these broadly as quantifying either the intrinsic dimensionality, the variability, or the regularity, periodicity, or self-similarity of the data from a nonlinear dynamical systems perspective. We perform this analysis on a novel full-body motion capture dataset collected in out-of-the-lab conditions for a variety of indoor environments. This is a unique dataset with a large amount (over 24 h total) of data from participants behaving without low-level instructions in out-of-the-lab indoor environments. We show that reasonable complexity measures can yield surprising, and even profoundly contradictory, results. We suggest that future complexity analysis can use these guidelines to be more specific and intentional about what aspect of complexity a quantitative measure expresses. This will become more important as wearable motion capture technology increasingly allows for comparison of ecologically relevant behavior with lab-based measurements.

4.1 Introduction

In this chapter, we examine how different locomotion activities compare in terms of their relative complexity. We use different reasonable measures of complexity to rank activities and compare them with our own intuition about the complexities of activity. For example, given the following list of activities- 1) forward (unobstructed) walking, 2) backward walking, 3) sidestepping, 4) walking in congested spaces with obstacles and 5) meandering in open spaces with a few obstacles and people, most people would have some guess about the relative complexities of these activities (probably $1 < 2, 3 << 5 < 4$). Do the measures of complexity found in gait literature consolidate these intuitions or do they counter it? **Another goal is to examine whether walking in the natural uncontrolled environments really more complex than unobstructed forward walking (the most recorded and analyzed locomotion activity** (See Background: Motion capture and public datasets). Next, we discuss the motivation for this work. In section, 4.3 the methods are presented. Section 4.4 discusses the key results, some practical recommendations are made in section , and finally the limitations and challenges are discussed in section 4.6.

4.1.1 Motivation

Measurement of the complexity of motor output [73, 23] is a common and essential component of gait analysis. It can be used for basic science, providing a window into how the brain generates movement [117], performs sensation, and how neural control interacts with biomechanics [25]. It can also be used for clinical gait analysis, with real implications for prescription of interventions and functional classification e.g. [109]. For example, a decrease in motor output complexity might indicate a reduced ability to adapt to stresses [78, 7, 42]. According to this reasoning, decreased complexity could indicate a reduced capacity for rejection of variability [42], or a deterioration of the complex human rhythms of movement associated with healthy function [23]. It might also serve as a tool to examine how well the current techniques used in control of assistive devices approximate the natural human gait.

For example, it is well known that human gait exhibits variability (one measure of complexity) across strides due to several factors like environment or fatigue, while the control of assistive devices is often rigid and deterministic. Since different activities exhibit varying degrees of complexity, it may be that if a deterministic control technique works for one activity with low variability, it will not translate well to different, highly variable, activity. Thus, examining the variability (and more broadly, the complexity) of different activities, is needed.

In the past, gait datasets have been largely confined to in-the-lab environments. Most available gait data has been restricted to uncluttered level ground ambulation or walking on a treadmill. As a result, much of the analyses and conclusions about human gait are drawn from a limited context. For example, there are no previous studies that compare commonly recorded gait activities like forward walking in a straight line to daily unconstrained walking in public places in terms of their complexity. However, recent developments in wearable sensors have driven increased interest in measuring human movement under a more diverse set of activities and situations. This makes it possible to analyze and compare these activities with the most commonly analyzed activity: flat ground walking in a straight line.

It is actually not trivial to quantitatively measure and define the relative complexity of different activities [73]. From our natural experience of life, we understand that avoiding obstacles, navigating challenging terrain, or dealing with uncertainty in the environment should result in more complex movement. We also intuit that movement outside of a gait lab, in the presence of other people and a changing environment, should result in more complex movement. But what, precisely and quantitatively, does that mean? There are several reasonable quantitative measures of complexity that actually are measuring different aspects of the data, and *can be contradictory.*

In this chapter, we attempt to define reasonable boundaries for these questions, and demonstrate some experiments and measurements that begin to answer them. Our goal is to contribute to a standard practice of gait complexity analysis, and especially comparison of different activities, as movement studies increasingly take place in more natural,

unconstrained contexts. We present a multi-subject (See Table 4.1) full body kinematics dataset that captures diverse activities like forward walking, backward walking, side stepping, avoiding obstacles by stepping over them, navigating around obstacles in structured and controlled environments as well as unstructured and uncontrolled natural environments, and stair ascent and descent. We qualitatively and quantitatively compare these activities to straight-line forward walking.

First, we provide background and context for complexity analysis in Section 4.2. We also present the potential contradictions in different complexity measures using a toy example. In Methods (Section 3), we describe the experiment, data analysis details, and quantitative outcome calculation methods. In Results and Discussion (Section 4) we present comparisons of the relative complexity of the different activities, and consider the importance of these outcomes, especially when different notions of complexity result in apparent differences. Finally, we discuss some limitations of our analysis.

4.2 Background: Complexity analysis

Previous analyses of complexity may be generally categorized as being inspired by three notions [73, 23]: 1) dimensionality, 2) variability, and 3) nonlinear dynamics. Here we use measures from each of these. As we describe in the Results and Discussion, there can be important differences in the apparent complexity of gait depending on the specific measures being used.

4.2.1 Complexity in terms of dimensionality

This approach assumes that the greater the number of dimensions (degrees of freedom) required to describe the data, the greater the complexity of the data [73]. A common method used to capture dimensionality of the data is Principal Component Analysis (PCA). Dimensionality is defined as the number of principal components required to capture a certain level of variance in the data. The greater number of PCs required to explain the desired level of variance in the data, the greater the complexity of the data. There are a variety of

other matrix factorization algorithms are used to identify underlying regularities or synergies in movement data [110]. As we have shown in other work, there are other advantageous nonlinear methods of identifying the underlying dimensionality [11, 80]. However, the most straightfoward method commonly used in the current gait literature is PCA [73], so we will constrain ourselves to that measure for this analysis.

4.2.2 Complexity in terms of variability

An alternative way to measure complexity is to assess the amount of deviation in a signal. For example, the Standard Deviation (SD) or Coefficient of Variation are common measures that use this approach [73]. This allows the complexity of even very low-dimensional data to be quantified meaningfully. For multi-variate data the determinant of the covariance matrix, also known as Generalized Variance [125], can be used a variance measure. Another measure of variability (GaitSD) has been proposed in [96], to measure the variability of gait waveforms across strides. Larger variability implies greater complexity under these definitions.

4.2.3 Complexity in terms of non-linear dynamics

Tools from non-linear dynamical system theory have been used to measure the regularity and periodicity of gait signals across time. Decker et al. (2010) [23] describes two kinds of analyses: State space examination and self-similarity evaluation, used to assess gait complexity.

State-space examination is done using the Largest Lyapunov Exponent (LyE) and Correlation Dimension. The LyE measures the average exponential rate of separation of neighboring trajectories of the attractor, while Correlation Dimension is a measure of the fractal dimension of the attractor. A positive LyE indicates aperiodic signals while a negative or zero LyE are associated with periodic signals. Random data are generally characterized by a large Correlation Dimension and LyE values while deterministic (periodic or chaotic) data exhibit smaller values.

Self-similarity evaluation is done to examine the presence of repeating patterns in the gait signal. Entropy based measures like approximate entropy (ApEn), sample entropy (SampEn), detrended fluctuation analysis (DFA) and multiscale entropy (MSE) are used to this end [22].

4.2.4 Contradictions in measures of complexity

These are all reasonable, but potentially contradictory, quantitative measures of complexity because they are measuring different characteristics of the data. We can understand this from the following 2D toy example in Fig 4.1 as follows: In the first row, we see two Gaussian data clouds which we can imagine as being generated by two different activities. Consider if we define variance of the data as the measure of complexity. We could reasonably use Generalized Variance for multi-dimensional data [125], which is defined as the determinant of the covariance matrix (Σ). We would rank the red cloud to have greater complexity, because the red cloud implies people need to attain a broader range of distinct states with their body. On the other hand, if we use dimensionality as the measure of complexity, we would not be able to distinguish between the two activities, as both equally employ the 2 available degrees of freedom.

In the second row, we see that a dimensionality measure would rank the blue activity to be more complex, since the red activity seems to be generated by a single independent factor, while the generalized variance would rank both the activity clouds to be of similar complexity.

These examples highlight that we need to exercise caution when discussing complexity. Although it appears to be a concrete and quantitative concept, it is necessary to be more specific about what kind of complexity we are measuring. In the toy example, simple visualization of the data helps to provide an intuitive grounding, but as we analyze time series data from many sensors simultaneously, we cannot rely on intuition. In the remainder of this chapter, we will demonstrate this concretely using five standard complexity measures.

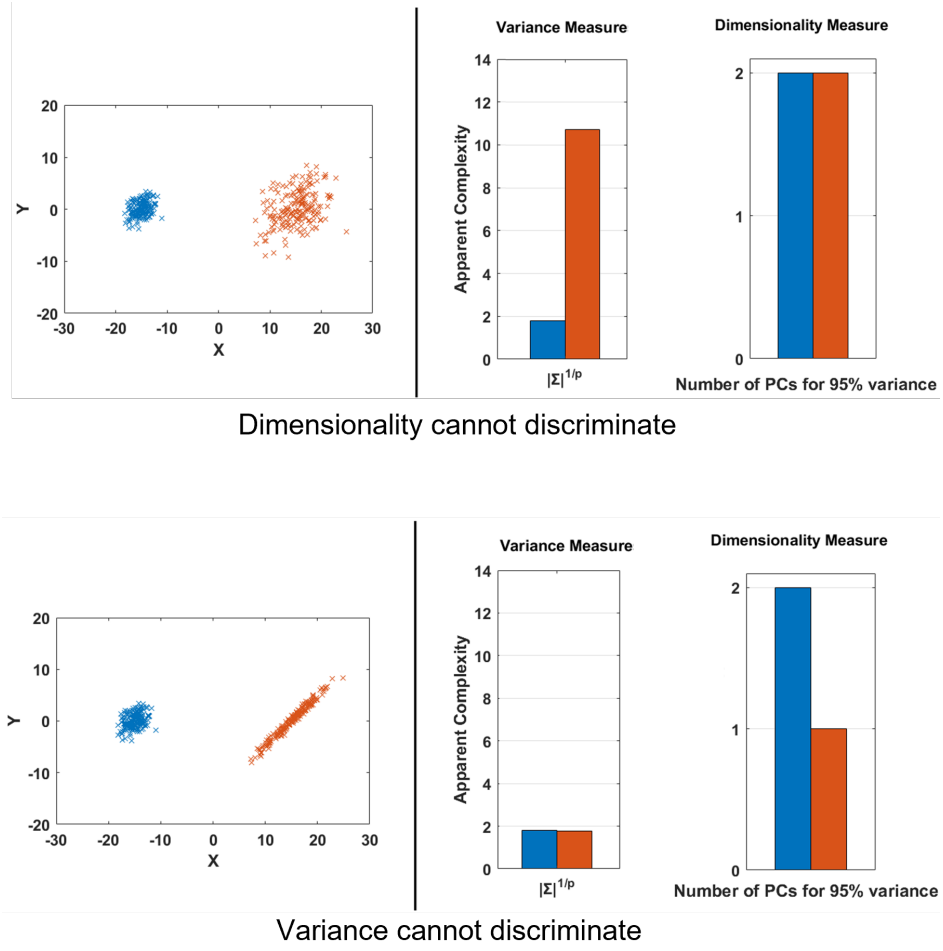

Figure 4.1: Toy Example to demonstrate some cases when different measures of complexity can fail to discriminate two distinct datasets and lead to contradictory outcomes. This example deals with only variance and dimensionality, but similar parallels exist for the other measures of complexity, such as stability from a nonlinear dynamics perspective.

4.3 Methods

4.3.1 Experiments and Subjects

For each data collection session, the subject was briefed about the experiment and informed consent was obtained. All activities were approved by the Institutional Review Board at

University of Washington. The entire dataset will be made available on a public repository (https://github.com/abs711/The-way-of-the-future) and more details about the data are presented in [102] Subjects' joint kinematics were recorded using an Xsens Awinda full body motion capture system *(Xsens Technologies, Enschede, Netherlands)*, consisting of 17 body-worn inertial measurement units placed at each segment of the limbs, as well as sternum, sacrum, shoulder scapula, and forehead. After a system specified n-pose calibration, the software provides joint kinematics in a 3D environment. 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 [126]. Data were sampled at 60 Hz, from a total of 22 joints in 3 anatomical planes (sagittal, frontal, transverse) for each trial. The kinematics data were reprocessed using the 'HD' processing feature, provided by the manufacturer for offline use, to enhance quality and remove noise [74].

In this chapter we limit the complexity analysis to kinematics data from only the lower limb joints: hip, knee and ankle from sagittal, transverse, and frontal planes for both the limbs. Thus, a total of 3 anatomical planes from 6 lower limb joints were used in our analysis i.e. 18 degrees of freedom.

Subjects ambulated in a variety of ways, including walking, sidestepping without crossing legs, navigating through obstacles, making turns, etc. as they deemed necessary in order to navigate the environment. Their speed was self-selected and their path around obstacles was not instructed. The movement was performed outside of a laboratory, in the corridors, indoor rooms and atrium of a building. The architecture for one of the classrooms and the atrium is shown in Figs. 4.2 and 4.3. The dataset was manually parsed into six activities for analysis. The activities that were parsed out for complexity analysis are: 1) **Forward walking** (straight line), 2) **Backward walking** (straight line), 3) **Left sidestepping**, 4) **Right sidestepping**, 5) **Navigating in classrooms**, and 6) **Navigating in an atrium**.

The numbers of participants and demographic information for each of the activities are shown in Table 4.1.

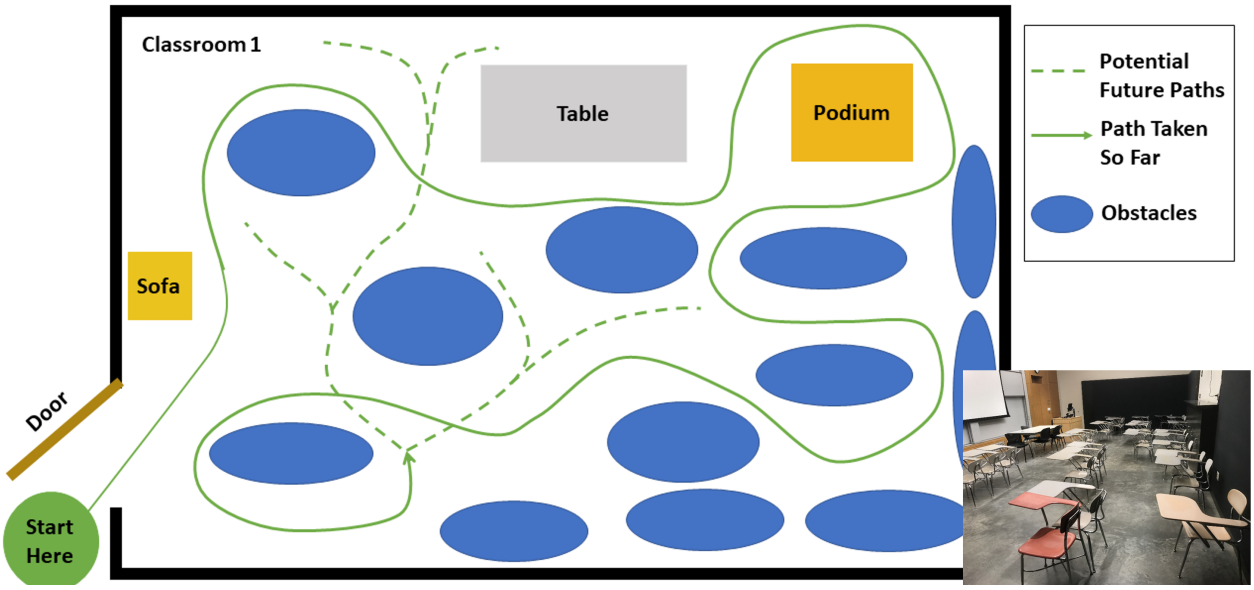

Figure 4.2: **Classroom:** Architecture of one of the classrooms. The arrangement of obstacles was not controlled and varies across subjects. The subject walked at self-selected speed and along self-selected path, The experimenter directed the subject to change their path only if the subject repeated the same path more than 2 times.

4.3.2 Data Analysis

In the Background: Complexity Analysis section above, we described three major notions that can be used to analyze complexity: Dimensionality, Variability, and Nonlinear Dynamics. We used measures related to these notions as described below, to analyze the complexity of activities.

Dimensionality The dimensionality of an activity is defined as the number of PCA principal components required to explain 95% variance in the activity ($N_{95\%}$). We used the function 'pca' from the Statistics and Machine Learning Toolbox. MATLAB 2020b for the analysis. $N_{95\%}$ was computed using the matrix of 18 dimensional time series from each trial. The mean and standard deviation of $N_{95\%}$ across trials are reported for each activity.

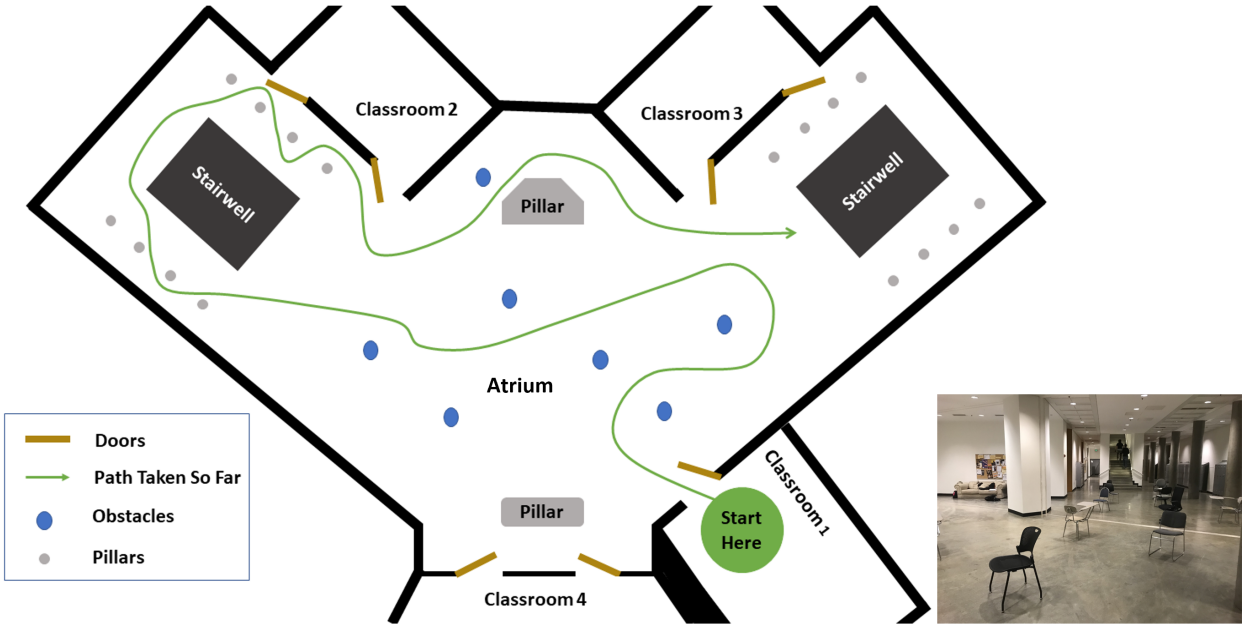

Figure 4.3: **Atrium:** Architecture of the Atrium. The arrangement of obstacles was not controlled and varies across subjects. The subject walked at self-selected speed and along self-selected path, The experimenter directed the subject to change their path only if the subject repeated the same path more than 2 times.

Variability We examined variability according to two different measures. The first is the Determinant of the data covariance matrix. It is not in standard use for human movement analysis, but it is a longstanding way to quantify variance in multidimensional data [125].The second measure is GaitSD, which measures how variable the gait cycles are from one another [96]. GaitSD is described in equation 4.1.

Table 4.1: Activities and Subject Details

Activities	**# subjects**	**Age(yrs)**	**Height (cm)**
Forward walking	9 females; 11 males	26.2 ± 2.7	174 ± 10.9
Backward walking	4 females; 5 males	21.5 ± 2.4	173.4 ± 6.9
Sidestepping	4 females; 5 males	21.8 ± 2.2	172.8 ± 6.8
Classrooms and Atrium	12 females; 11 males	22.8 ± 2.7	171.2 ± 9.7

$$
\begin{aligned}
X_{ij} &= i^{th} \text{ gait cycle defined over T time instances, T= 101} \\
X_j &= \frac{1}{N}\sum_{k=1}^{N} X_{ij} \text{ , } N = \text{number of gait cycles} \\
GVSD^2 &= \frac{\sum_{j=1}^{T}\sum_{i=1}^{N}(X_{ij}-X_j)^2}{T(N-1)} \\
GaitSD &= \sqrt{\frac{1}{p}\sum_{k=1}^{p} GVSD_k^2} \text{ , p (number of joints) = 18.}
\end{aligned}
\tag{4.1}
$$

Gait cycles were determined using the foot contact data provided by Xsens, and all the joint angles were time normalized to 101 points using the MATLAB command- *'interp1'*.

Nonlinear dynamics Following methods from [23, 14], we used the Largest Lyapunov Exponent (LyE), and Multiscale Entropy (MSE). These measures were calculated using ankle, knee, and hip kinematics in the sagittal plane.

To calculate the LyE, we first reconstructed the state space from one dimensional time series (sagittal ankle, knee, and hip separately), using Takens' theorem [75]. The delay for reconstruction was estimated using Average Mutual Information (AMI) [35]. It was set to be the first local minimum of AMI. Embedding dimensions were determined using Global False Nearest Neighbors (GFNN) analysis [53]. Embedding dimension was set to the minimum value that satisfied percent false nearest neighbour less than 10%. LyE were then determined using MATLAB's Predictive Maintenance Toolbox. The package calculates LyE using the

algorithm developed by [94].

Multiscale Entropy is a way to analyze the self-similarity of a one dimensional time series. There are multiple ways to calculate MSE [46], but here we use a robust variant, Composite multiscale Entropy (CMSE), proposed in [127]. The Complexity Index (CI) is defined in equation 4.2. m and r were chosen as 2 and 0.2 respectively in accordance with [10] and values of τ ranged from 1 to 20.

$$CI = \sum_{\tau=1}^{N} CMSE(x, \tau, m, r) \tag{4.2}$$

τ = time scale index, N (Total number of time scales) = 20

4.4 Results and discussion

For each of the activities, we calculated the complexity measures of Dimensionality (Figures 4.4 and 4.5), Variability (Figure 4.6), and Nonlinear Dynamics (Figures 4.7 and 4.8). For each of these, we report the relative complexity of the activities and discuss when the results are contradictory or unexpected.

4.4.1 Dimensionality: Sidestepping is the most complex activity?

Figure 4.4a shows variance explained across subjects, from PCA of Lower Limb (18 dof) for the different activities. The number of principal components required to explain 95 % of the variance ($N_{95\%}$) is shown in figure 4.4b. We observe that left and right sidestepping require the most components to explain the variance, while forward walking requires the fewest. Complexity analysis in terms of dimensionality as measured by PCA, then, concludes that sidestepping is the most complex activity while forward straight line walking is the least. Dimensionality is appealing as a measure of complexity because it aligns with the intuition that a "more complex" task should require more independence among its degrees of freedom. Dimensionality has been successfully used in gait analysis and has aligned with clinical notions of mobility and the scientific notion of synergies [61, 109, 93]. However, in this

study we show that using PCA and "variance accounted for" yields counterintuitive results. Although forward walking is measured as least complex, left and right sidestepping arise as the most complex, while navigating freely amongst challenging obstacles, as in the classroom activity, is measured as less complex than unobstructed sidestepping.

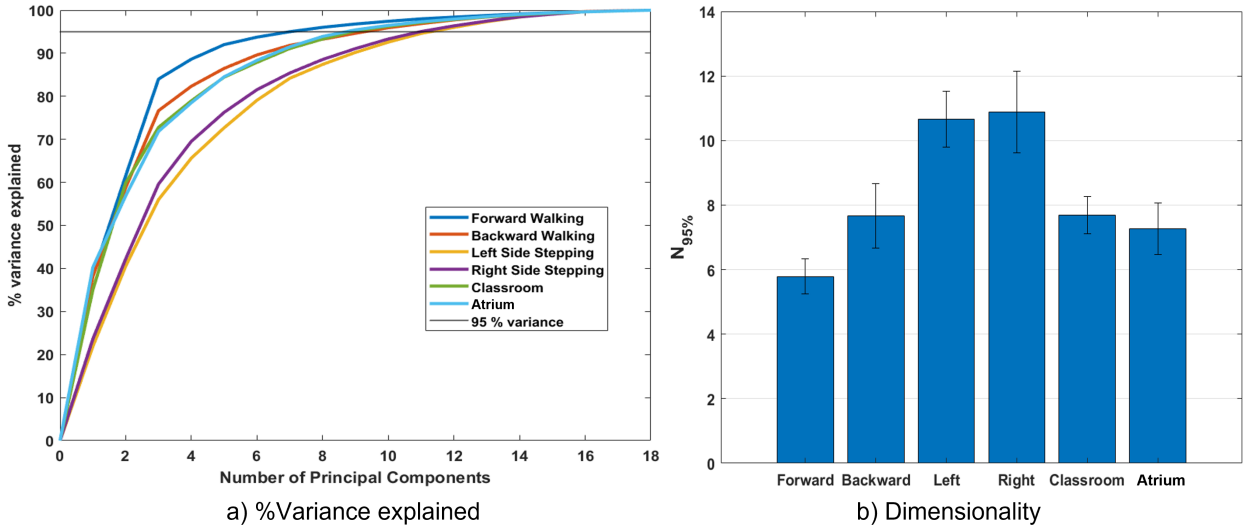

Figure 4.4: a) **Percent variance accounted for** by each principal component and the sum of the first n principal components (line plots), for different activities. These were calculated using the data from all subjects. b) $N_{95\%}$ values for all the activities. $N_{95\%}$ is the number of principal components required to explain 95% variance in the data from each subject. $N_{95\%}$ indicates all other activities have higher dimensionality, and therefore complexity, than forward walking. Surprisingly, this metric indicates that left and right sidestepping are more complex than walking in a natural environment.

This result is surprising because from our experience of life, we understand that avoiding obstacles, navigating challenging terrain, dealing with uncertainty in the environment, etc. should result in more complex movement in Classrooms and Atrium. It should require us to use more degrees of freedom to navigate. This result can be interpreted in two mutually exclusive ways: 1) Even though sidestepping and backward walking are expected to be highly

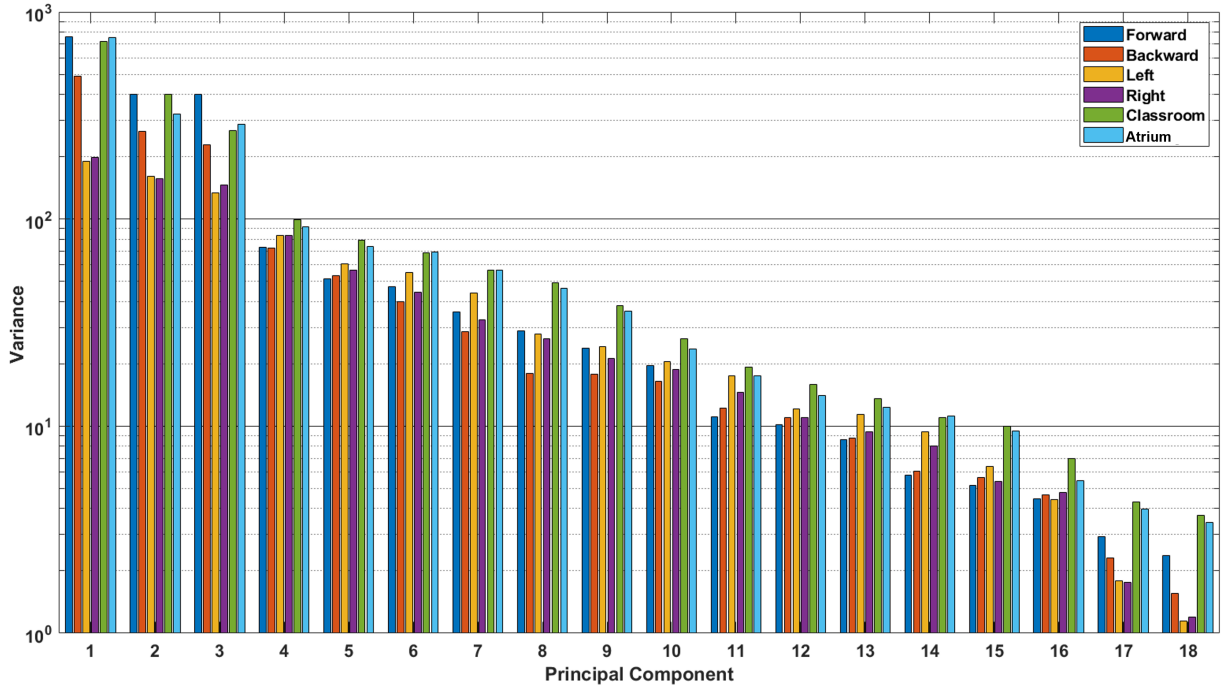

Figure 4.5: **Absolute variance accounted for** by each principal component, for different activities. These were calculated using the data from all subjects. We see that the last few principal components for Classroom and Atrium show considerably larger amount of variance than sidestepping, even though they are ignored by PCA when measuring dimensionality (see Fig. 4.4).

repetitious, they are less practiced, and thus show less coordination between joints. Thus, the data has more degrees of freedom than expected. 2) Alternatively, the result could be interpreted to indicate that PCA shouldn't be used to measure and compare dimensionality when the two datasets have different overall absolute variance (See Fig. 4.5). For example from Fig. 4.4b, we see that Sidestepping has a dimensionality of approximately 11. Now, from Fig. 4.5, we see that 11^{th} PC for Classrooms and Atrium has greater variance that for Sidestepping activities, but is ignored when 95% variance is used as the criterion to decide the dimensionality of data. This highlights the need for further examination of our intuition

about the complexity of locomotion activities, and to be aware of these issues when using PCA for measuring the dimensionality of activities.

4.4.2 Variability: Classroom walking or Backward walking is the most complex activity?

Figure 4.6a shows the generalized variance for the different activities. Forward walking shows the smallest generalized variance indicating tighter coordination of joints, while Classroom shows the largest value, indicating more variability in joint angles and less coordination amongst them. Figure 4.6b shows that backward walking has the largest GaitSD, indicating greater stride to stride variability of joint kinematics.

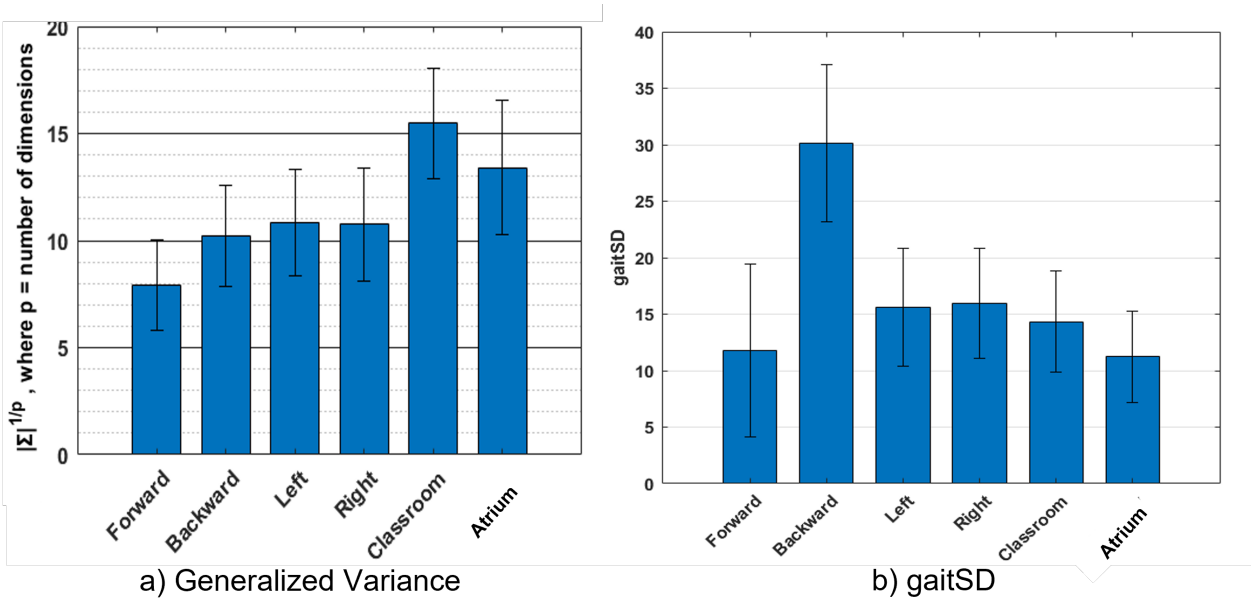

Figure 4.6: **Variability**: We use two different measures of variability- a) Generalized Variance (geometric mean of the variances along the Principal Components) which measures the spread of the multi-dimensional data. Walking in classroom exhibits greater complexity in the joint angles, than other activties according to this metric, b) GaitSD which measures variability of gait kinematics across strides, ranks backward walking to be of greatest complexities. The values reported are inter-subject mean and standard deviation.

Variability is a perfectly reasonable way to quantify the complexity of data. While PCA uses variance and covariance to measure complexity, it only looks at how variance is distributed across different dimensions i.e. relative (or percent) variance. It can be instructive to look at absolute variance as well. Here we use generalized variance and GaitSD to measure variability in two distinct ways. Generalized variance is a measure of how much volume in the state space is occupied by a given activity. In other words how many different configurations of joints are achieved by a given activity. Navigating the Classroom and the Atrium must be expected to show greater generalized variance than other repetitious activities, because they require extemporaneous movements to avoid obstacles, change directions, etc. Generalized Variance comes out to be highest for those activities, matching our expectation.

GaitSD measures the variability of joint angles across gait cycles. It is not sensitive to the amplitude of joint angles (and thus the volume occupied in the joint-space) but instead the deviations at different phases in a gait cycle from the mean gait cycle. In other words, trying to do a repetitious activity but failing to do it exactly would have a greater GaitSD value than doing many kinds of movements but with more precision. This might explain why backward walking has a greater GaitSD value than other activities. Backward walking is presumably less practised in daily life than the other activities. Sidestepping also shows slightly higher values than other regularly practised activities like forward walking, walking in the classrooms and atrium.

This highlights that variability can be measured in different ways but more importantly the different measures need not agree. GaitSD, a measure of gait consistency, rates backward walking to have almost twice the amount of gait variability than unrestricted classroom walking. Further examination is required to understand the mechanisms leading to this observation, because naively we would expect unrestricted classroom and atrium walking to have greater variability than backward walking which is expected to be repetitious.

4.4.3 Nonlinear dynamics: Classroom or Sidestepping is the most complex activity?

Figure 4.7 shows the LyEs for different activities. For the most part, the activities show positive LyE values indicating non-periodic gait signals. Classroom and Atrium show largest LyE of all the activities, across all the joints. For the knee joint, LyE of sidestepping for some subjects are negative or close to 0, while other subjects have large positive LyE (close to 6). This might be an artifact of noise in the data and needs more investigation. Figure 4.8 shows the analysis of ankle, knee and hip joint trajectories, using MSE, computed using sample entropy (S_E) over 20 time-scales.

Human gait can be modelled as a dynamical system. Non-linearity in dynamical systems leads to different kinds of complexities than the ones we analyzed above. This has to do with periodicity, regularity and predictability of temporal dynamics of the system. We used LyE and Multi-scale Entropy to analyze complexity from this point of view.

LyE measures how quickly neighbouring trajectories in the dynamical system converge (negative values) or diverge (positive values). Larger positive values indicate faster divergence and thus lesser predictability of gait further into the future. Since, Classroom and Atrium exhibit largest values, they should be expected to be less predictable. This is expected because navigating obstacles would require significant deviation of gait from the immediate history, thus less predictability into the future.

Multi-Scale Entropy measures how many repeating patterns are there in a signal over different time-scales. Intuitively, it measures the regularity, or predictability, of a signal. We see that unusual activities i.e. sidestepping and backward walking show more irregularity in gait than more common activities i.e. forward walking, Classroom and Atrium. This might be a result of lack of practice in sidestepping and backward walking.

Once again we find major disagreement between the two measures used in this analysis, highlighting contradictions between different types of complexities in temporal dynamics.

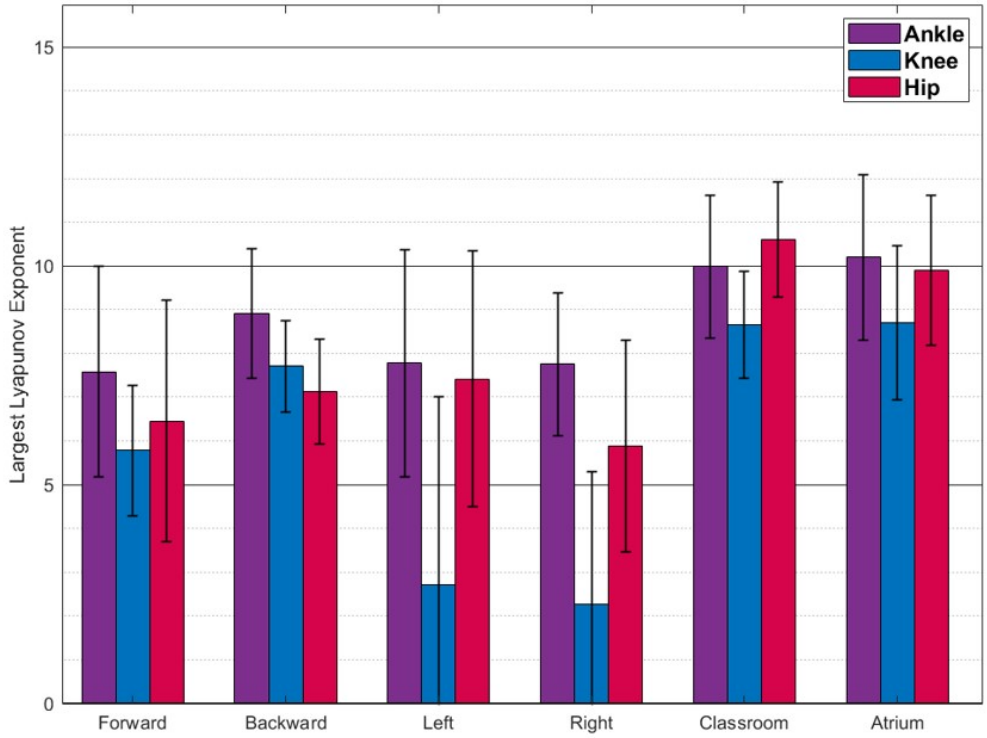

Figure 4.7: **Largest Lyapunov Exponent(Mean ± SD)** The values reported are mean and standard deviation across the trials from all the subjects. Walking in classrooms and atrium shows greater complexity than other activities.

4.4.4 Complexity cannot be defined as a unitary concept

These results demonstrate that there are several ways to measure different aspects of data complexity. These measures often do not rank activities similarly. For example, dimensionality as measured using PCA ranks sidestepping to be the most complex, but gait cycle variability as measured using GaitSD ranks backward walking to be the most complex, while divergent nonlinear dynamics as measured using LyE ranks atrium as the most complex. Looking forward for practitioners of movement analysis, *complexity* should probably

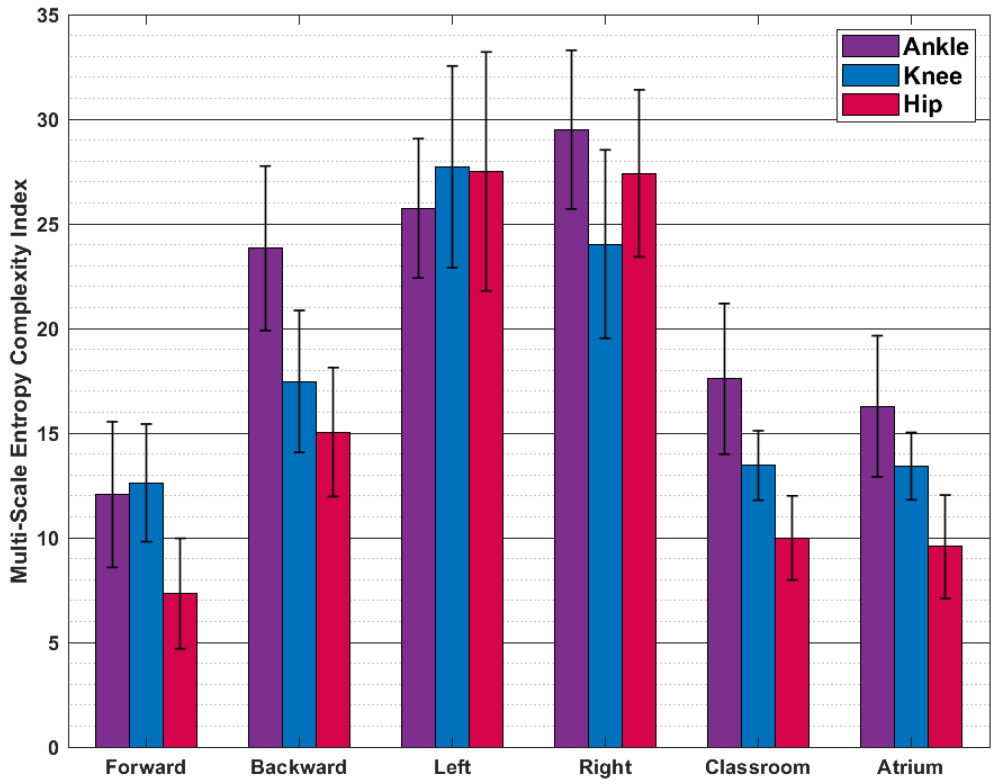

Figure 4.8: **Multiscale Entropy (Mean ± SD)** The values reported are mean and standard deviation across the trials from all the subjects. Sidestepping shows greater irregularity and thus complexity, than other activities.

be avoided as a single concept in favor of specific measures. For example, when we use PCA analysis, we should state that we are measuring degrees of freedom, not accounting for the scale of the variance. We summarize the rankings of complexity in Figure 4.9. As can be seen, no column, corresponding to activities, is agreed upon in complexity ranking by the different methods.

Figure 4.9: Overview of how each complexity measure ranks the six activities. While there are some similarities, it can be seen that each measure is sensitive to different characteristics of the complexity of the data, and that many of the results are surprising or counterintuitive.

4.4.5 Forward walking is the least complex activity

Most of the measures agreed on forward walking being the simplest activity. Although GaitSD and LyE didn't strictly rank it as the least complex, it is very close, as can be seen in Figures 4.6 and 4.7. This is expected since forward walking is highly practiced and repetitious, and doesn't involve deviations to account for obstacles.

4.5 Practical recommendations

- PCA ranks sidestepping to be more complex and backward walking to be as complex as walking around obstacles in classrooms and atrium. This is counterintuitive. On further analysis, we found that sidestepping doesn't necessarily have more variance in the last PCs than classroom and atrium, as can be seen from figure 5. This can be understood from the 2D toy example, as shown in Figure 1, bottom row. As can be seen, even if the minor principal component has the same variance for both blue and

red clouds, PCA would rank the blue cloud to be more complex than red cloud, because it ignores the absolute variance and only accounts for relative variance. Thus, we need to account for absolute variance, before we use PCA to rank the dimensionality of different activities. To measure the absolute variance, we recommend that researchers use Generalized Variance.

- Usually, in the gait literature, variance is used to analyze one-dimensional signals. In our analysis, we used Generalized Variance as a measure of absolute variance for multi-dimensional data. We found that the resulting complexity ranking of the activities aligned well with our expectations. Thus, we recommend using Generalized Variance to measure the scale of the data.

- In our analysis, we found that Largest Lyapunov Exponent values to be quite different from [16]. This could be attributed to sensitivity of the measure to noise in the data or the length of the data. In addition, computation of Largest Lyapunov Exponent assumes a time-invariant and autonomous dynamical system. Thus, we recommend against the use of the measure, unless the accompanying assumptions are tested for.

- In the gait literature, complexity is an umbrella term that measures different aspects of the data- dimensionality, variance and nonlinear dynamics. Since these measures do not always agree, we recommend against the usage of the term ‘complexity‘ and instead using the terms that emphasize the metric being used e.g. dimensionality.

4.6 Limitations

Dimensionality, as a concept used in mathematics, is much broader than we are using it here for gait analysis. For example, dimensionality can be defined as the number of Euclidean dimensions, topological dimensions [111], fractal dimensions, etc. We only use number of principal components as it has a precedent in gait analysis. Even in terms of integer dimensions or degrees of freedom, other dimentionality reduction techniques like autoencoders

could be used to estimate dimensionality [80, 11].

Calculation of the Largest Lyapunov exponent requires the assumption that the system is *autonomous,* and time invariant [98, 94]. This assumption could be broken by learning effects, fatigue, etc. Additionally the Largest Lyapunov exponent requires large amounts of data to be confidently calculated. So, care must be taken to ensure that adequate data sizes are used. It has been shown that accurate Lyapunov dimension calculation requires hundreds of gait cycles, and can be sensitive to preprocessing choices, such as using a fixed number of strides or a fixed number of data points [48]. When comparing activities that have very different total amounts of data, or different standards for preprocessing, care must be taken for this measure to be meaningful. This factor is not limited to Lyapunov dimension for gait; some measures, such as those used in heart rate variability estimation, have been shown to require small data sizes, while others require more data for robust estimation [20].

Since the data collection process is time consuming, any particular participant could not perform all of the different activities. While there is no missing data from any particular participant, each performed only a subset of the possible activities, as shown in Table 4.1. As a result, the analyses we present here cannot account for individual differences in complexity. Individual gait characteristics could be practically important, for example in designing assistive devices, and should be accounted for also.

The data were also measured for a narrow age range of young people indoors, in an experimental session. We anticipate that their movement was more reflective of their natural patterns for those environments compared to being in a gait analysis laboratory. However, there were still factors that could produce "demand characteristics" [95]. These are changes in behavior due to expectations, whether conscious or not, of the purpose of the experiment or increased conscious control over normally unconscious movements.

Wearable motion capture provides a convenient and versatile means to record movement without instrumentation of the space, but it also is sensitive to challenges in calibration, placement of markers, and precision of recording. There are degrees of freedom with less range of motion that are nonetheless important biomechanically, such as knee and ankle

frontal plane, that are measured with less validity than gold-standard marker-based tracking systems.

This analysis does not include statistical significance testing. We have calculated the common complexity measures and reported their mean and standard deviations where appropriate, or other commonly used reports such as percent variance explained in Figure 4.4a. The large differences or similarities are apparent to see the performance of these measures, but a more formal treatment could include statistical significance testing.

4.7 Conclusions

In this chapter, we examined the complexity of different human locomotion activities using various measures of complexity pertaining to dimensionality, variability and nonlinear dynamics. We find that most of the measures rank the most commonly analyzed activity, walking forward in a straight line, to be the least complex. More importantly, different measures disagree about the relative complexity of the remaining activities. Thus, defining complexity as a single notion is challenging and we might need to be cognizant of what aspect of the data we wish to analyze when using any particular measure.

Chapter 5

OPTIMIZING REPRESENTATIONS OF MULTIPLE SIMULTANEOUS ATTRIBUTES FOR PERSONALIZED GAIT GENERATION USING DEEP LEARNING

Abhishek Sharma, Eric Rombokas

Publication in review

Abstract

Rich variations in gait are generated according to several attributes of the individual and environment, such as age, athleticism, terrain, speed, personal "style", mood, etc. The effects of these attributes can be hard to quantify explicitly, but relatively straightforward to sample. We seek to generate gait that expresses these attributes, creating synthetic gait samples that exemplify a custom mix of attributes. This is difficult to perform manually, and generally restricted to simple, human-interpretable and handcrafted rules. In this manuscript, we present neural network architectures to learn representations of hard to quantify attributes from data, and generate gait trajectories by composing multiple desirable attributes. We demonstrate this method for the two most commonly desired attribute classes: individual style and walking speed. We show that two methods, cost function design and latent space regularization, can be used individually or combined. We also show two uses of machine learning classifiers that recognize individuals and speeds. Firstly, they can be used as quantitative measures of success; if a synthetic gait fools a classifier, then it is considered to be a good example of that class. Secondly, we show that classifiers can be used in the latent space regularizations and cost functions to improve training beyond a typical squared-error cost.

5.1 Introduction

Many applications in computer vision and robotics require generating novel human movement profiles. One such application is control of an assistive exoskeleton, for which we might need reference trajectories that vary according to desired speed, terrain, and the personal style of the user. Some previous approaches to producing trajectories are [91], [134], [45], and [92]. Some have included multiple attributes, for example speed and ramp terrain [91] and [92]. These methods use careful selection of basis functions, and optimize the error w.r.t. the training samples. This works well if a limited number of attributes are to be used to generate trajectories, and more importantly, when the attributes can be easily represented. However, with increasing availability of multi-modal datasets, there is a need for general-purpose methods that generate gait based on multiple attributes e.g. speed, personal style, terrain, mood etc. (Fig. 5.1a) There are also no general methods for prioritizing different attributes or to introduce new ones. In addition to reducing the need for hand-crafted features, these methods also need to maintain a degree of interpretability, which is essential for clinical applications. In other domains, eg. computer vision and audio synthesis, the notion

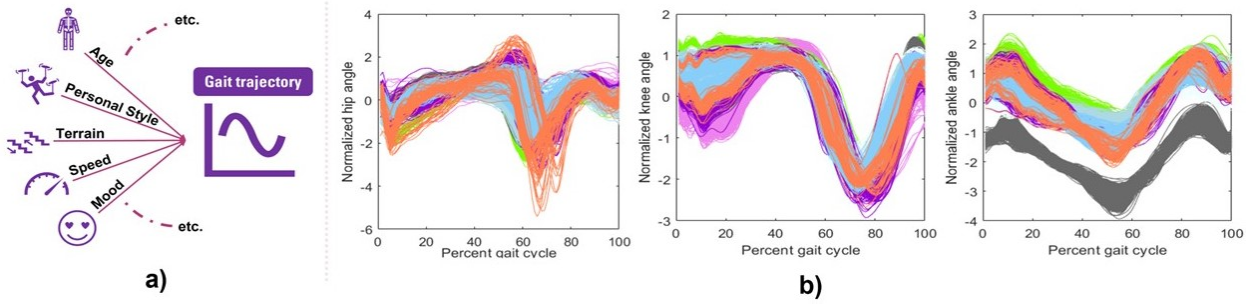

Figure 5.1: (a) Multiplicity of Gait Attributes, (b) Gait cycles of hip, knee, and ankle angles from 7 subjects at speeds ranging from 0.5 m/s to 1.8 m/s. Different colors represent different subjects. Note that each subject shows a distinct and largely consistent strategy for movement.

of *style transfer* has been introduced, where the idiosyncrasies or fundamental attributes of one example are combined with those of another to generate a completely novel sample. Fig. 5.2 shows an example of this, for the MNIST dataset, where digits are generated with desired handwriting styles. We adapt the method used for handwriting style transfer [65], for personalized gait generation at desired speeds, and show that can be made to work for this application domain. However, it doesn't allow us to simultaneously optimize for multiple attributes, and doesn't provide an interpretable way to split the problem into distinct subsystems.

To address these limitations, we show two methods that can be used individually or combined. The first is *cost function design*: multiple attributes may be simultaneously optimized for by adding terms to the cost function, and prioritized by varying their importance in the cost term. This allows for tuning the overall system to generate gaits that explicitly meet the multiple desired criteria. However, it doesn't allow us to guide the system internally to cleanly and distinctly create representations that capture the attributes. To address this, we demonstrate *latent space regularization*: the architecture is split into attribute-specific subsystems, each calculating targeted latent representations for a single attribute. Then these representations can be combined to solve the overall problem. For style transfer and trajectory generation, there are no well-accepted standards for quantitatively assessing success. Instead, researchers rely on visual inspection or on minimizing squared error with examples. The problem with focusing on error is that it doesn't necessarily capture the differences that distinguish examples from different attribute classes. For example, a synthetic image of a cat might have large error from any particular example image of cats, but still would be recognizable as a cat and not mistaken for a dog. Therefore we propose a quantitative success measure: classifiers that identify which group an example was drawn from. If a synthetic gait trajectory is correctly classified, it is considered successful in emulating that gait attribute.

5.2 Background

5.2.1 Multiplicity of gait attributes and style transfer

There can be multiple attributes that contribute to the generation of gait, from anthropometrics and body dimensions, to the environmental constraints, speed, to the somewhat ineffable personal style or mood of the walker. Generating gait from diverse attributes of gait can be challenging to do manually; especially for personal style which can be influenced by developmental history, injuries and other inexpressible event and thus hard to define.

We want to generate synthetic trajectories and we want them to be appropriate in multiple ways simulataneously. Individuality and speed are the most obvious examples. In the literature, there are methods for generating trajectories that are appropriate in a single way, usually in terms of low root mean square error (RMSE). What we are showing here is a general way to use representation learning for an arbitrary number of simultaneous constraints.

In machine learning, "style transfer" has been used to generate samples that meet multiple simultaneous criteria. Style transfer has primarily been used in computer vision for generating images. The content of one image can be recast into the style of another. Fig. 5.2 shows an application using an adversarial autoencoder [65] in rendering digits of the MNIST dataset in the style of other example digits. Each row in the figure has the same handwriting style, generated by a distinct individual. This technique was used to transfer a single attribute, dubbed to be "style", but in the current chapter paper we transfer both style and speed. We limit our analysis to style and speed in this chapter but the methods we present can be used to transfer an arbitrary number of distinct attributes, simply by including more attribute-specific encoders and terms in the cost functions (See section 5.2.3).

5.2.2 Quantitative Assessment of Generated Trajectory Quality

A person's style of walking is an important aspect of gait, perhaps even the most important component of prediction error in gait models [28], yet it is hard to measure and quantify.

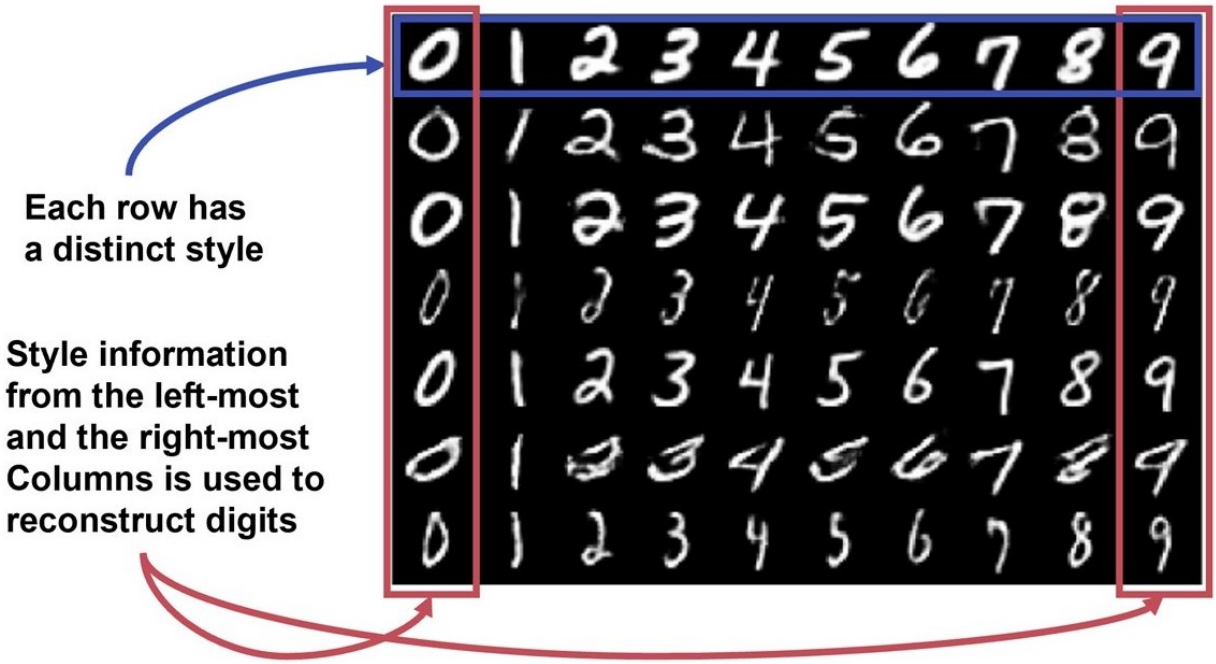

Figure 5.2: An example of style transfer in computer vision. Images of handwritten numbers (MNIST dataset) are used as example inputs. Given examples of handwriting style, synthetic handwritten numbers can be generated. Adapted from [65]

Fig.5.1b shows how different even relatively simple treadmill walking can be across subjects. To assess the quality of a generated trajectory, previous methods [92] use RMSE between the generated trajectory and a single representative trajectory or an ensemble of example trajectories. If a personalization method leads to reduction in RMSE, that is considered an indication of better personalization. However, the same RMSE can be achieved by two very different pairs of trajectories, potentially very different in their personalization success. For example, a trajectory with joint angles that briefly overshoot and then stabilize could have the same RMSE as one with a small bias that persists. These differences are potentially quite important to the subjective experience of personalization, but are not well expressed by RMSE.

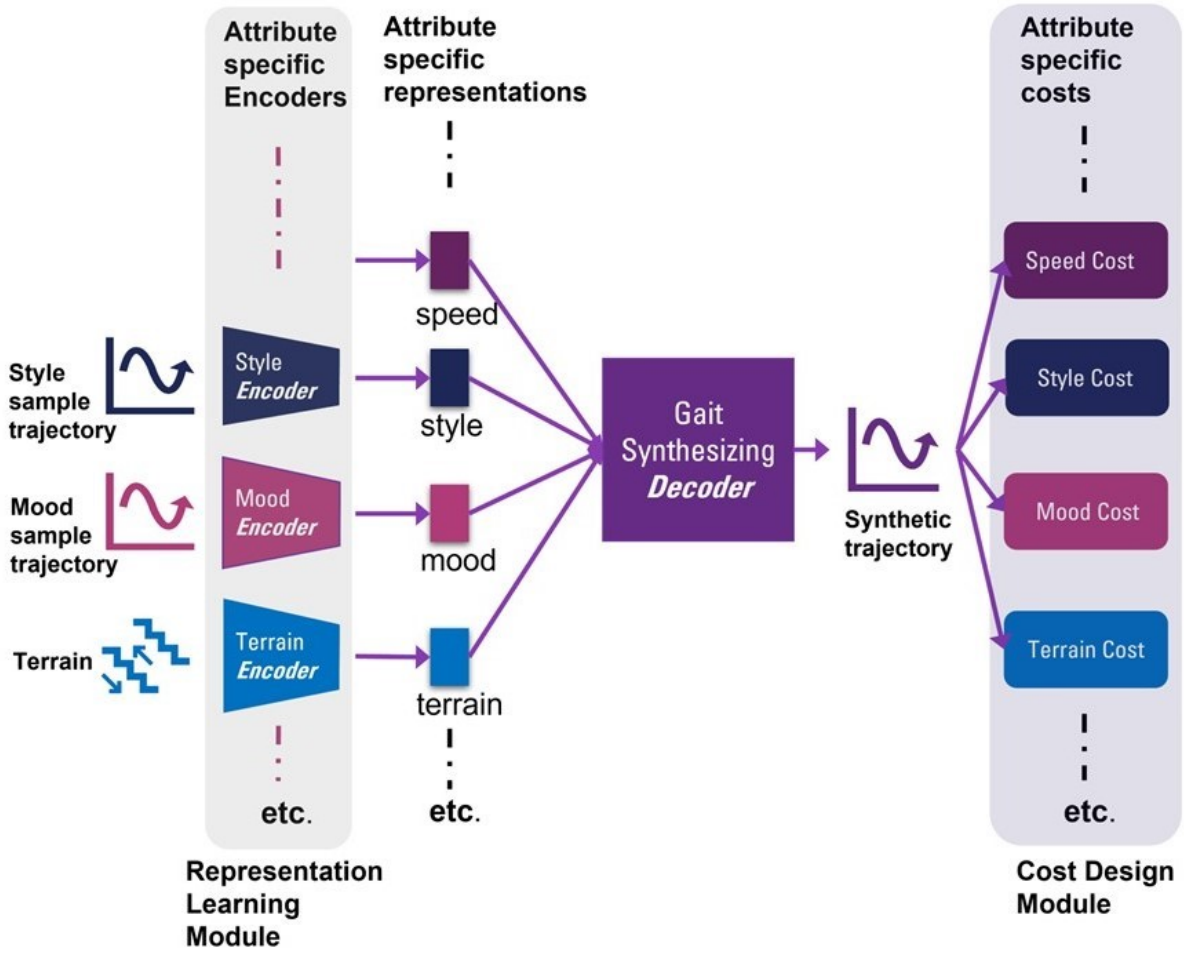

Figure 5.3: Concept diagram for generating gait with multiple desired attributes. Examples taken with different attributes, such as style, mood, or terrain, are encoded into attribute-specific representations. These latent spaces are learned, where that learning is encouraged to represent differences in that attribute and discouraged from representing differences in the other attributes. These representations are decoded to create synthetic trajectories. The decoder is learned using a multi-term cost function designed to encourage simultaneous transfer of each of the attributes.

To help address this, we propose a quantitative method based on training classifiers on the example data. The success rate of the classifier can be considered the ceiling for how discriminable the classes are. If a generated trajectory is sufficiently similar to the desired person or speed, it will fool that classifier, and this can be quantitatively measured.

5.2.3 Optimizing trajectory synthesis via optimal representation learning

The method used for handwriting style transfer (Fig. 5.2) can be directly adapted to generate gait, where instead of handwriting style, we synthesize walking style, and instead of which digit, we synthesize trajectories for different speeds. However, this straightforward adaptation of the method doesn't allow us to have much control over the output of the network. Firstly, minimizing RMSE doesn't necessarily lead to best person and speed appropriate performance. Secondly, the representations learned are not necessarily encoding the intended attributes (personal style in this chapter). To address this, we split gait generation into two distinct components: learning appropriate representations, and composing the learned representation to generate trajectories. Thus optimization can be applied at both stages (Fig. 5.3): at the output generation stage using cost function design, and at the attribute encoding stage using latent space regularization.

Cost function design Training gait models by minimizing RMSE doesn't necessarily promote attribute specificity in the output of the model, and might not be the best indicator of clinical suitability of the generated gait. The same RMSE can be achieved by two very different trajectories, and neural network convergence to lower RMSE doesn't necessarily mean that the solution is adequate. For example, certain phases of gait are more crucial to get right than others e.g. heel strike or toe-off. Therefore, constraining model outputs in more desirable ways by adding more attribute specific cost could be beneficial. In this work, we penalize the outputs of the model by using person and speed specific classifiers pre-trained on real trajectories from the training data. This approach could be extended to include cost terms for other desirable attributes. These cost terms can then be weighed to

emphasize one attribute in the output than other. For example, a diffusion-model approach to creating human motion for animation used a custom foot contact loss term to constrain realistic foot-ground interactions [114].

Latent space regularization Learning appropriate representations is crucial for successful machine learning systems. Not only can it help achieve better performance, but can help in interpretation of the results and in diagnosing model outputs. This is even more crucial when it comes to generative models. There are several examples in machine learning of regularization to achieve desired latent representations. For example, variational autoencoders enforce a Gaussian distribution on the bottleneck. This prevents fractured representation in the latent space and allows the decoder to create more plausible samples. Adversarial autoencoders use adversarial training to enforce any desirable distribution on the latent space. In this chapter, we show the benefit of controlling the latent representation for the task of generating personalized gait trajectories. We do this by not only promoting the desired attribute (i.e. style) in the latent space but also penalizing it from encoding speed information. This helps achieve drastic performance gains over the other approaches that don't explicitly control the latent representation.

5.2.4 Using Classifiers in Two Distinct Ways

It should be noted that this research is using classifiers in two distinct and independent ways. The first is as a quantitative measure for person and speed specificity of the synthetic trajectories (Section 5.2.2). These classifiers need to be able to classify real trajectories in terms of person and speed with a high degree of accuracy, and are trained on all the data from all the subjects. The second is in training: differentiable classifiers are used to regularize the latent representations or as terms in the cost function, to enforce desired attributes in generated trajectories, as described in Section 5.2.3. These classifiers are trained using the training data only to prevent information leakage from the test data during training.

5.2.5 Personalization in practice: how would this be used?

Fig. 5.4 describes how the personalization process (Fig.5.3) could be used. Envision a lower-limb assistive device such as a prosthesis or exoskeleton, that needs to generate personalized gait at several desired speeds. Gait samples from the target user are recorded, at a few selected speeds for which the assistive device has been tuned as per current clinical practice. These trajectories are then mixed with a databank of gait samples for a wide range of individuals and speeds. The resultant dataset is used to train neural network based encoder-decoder models (See Figs. 5.5-5.7). Speed information about the input trajectories is used to disentangle speed and style information. This allows the encoder to learn to extract personal style from input trajectories. The decoder learns to compose style and speed to generate trajectories of desired style and speed. Thus, we can extract style from a few optimized examples of the target user and combine them with desired speeds to interpolate and extrapolate across a wider range of speeds.

5.2.6 Contributions

- We demonstrate that classifiers can be used as a quantitative metric, beyond the more generally used RMSE, to evaluate the person-specificity and speed-specificity of a generated gait.

- We present a style transfer based framework for personalization of assistive devices. Style is extracted by a neural network encoder from samples of human gait at known speeds and used by a decoder to generate personalized gait at a variety of desired speeds.

- We use two generalizable regularization methods, cost function design and latent space regularization, to optimize the learned representations. These methods are more interpretable and customizable than a baseline style transfer method, and yield improved synthetic trajectories.

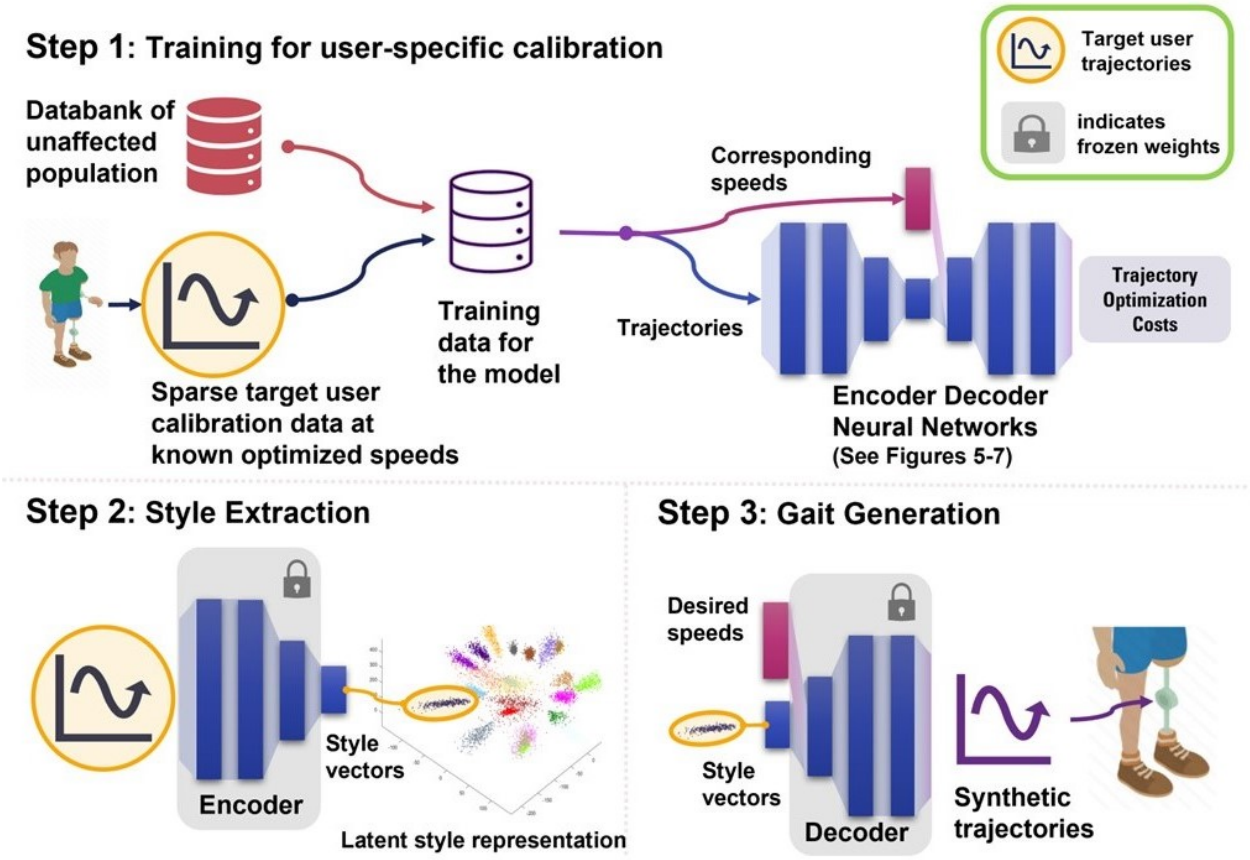

Figure 5.4: How the personalization system would be used in practice. Step 1: gait examples are recorded for a sparse set of speeds, tuned according to current clinical practice for eg. slow, medium, and fast walking. These data, plus the databank of gait examples, form the training data for the model. Training results in a model that is well calibrated to the target user. Step 2: User-specific style representations are pre-extracted to be used at run-time. Step 3: Style representations and desired speeds are input to the decoder to create a combined user-and-speed-appropriate synthetic trajectory.

5.3 Methods

5.3.1 Dataset

We used the publicly available dataset from [17]. The data were recorded from 22 able-bodied subjects walking at various speeds. In this chapter, we only consider the data from treadmill walking and use the data at 14 speeds 0.5 m/s to 1.8 m/s at an increment of 0.1 m/s. The resultant dataset has a total of 9073 trajectories. We selected 2 subjects at random as target users, and use the data from the remaining 20 subjects to form the "databank of unaffected population" (See Fig. 5.4).

5.3.2 Data Processing

The right hip, knee and ankle kinematics data at desired speeds was extracted from the MATLAB files available on Camargo et al. The kinematics data was segmented into gait cycles and time normalized to 100 points. Thus each gait sample is in a $R^{100\times3}$ space, where 100 is the number of time points and 3 is the number of joints. Each joint is normalized using its mean and standard deviation across all the training data. Each input gait sample is flattened into a R^{300} vector, for the neural networks to process.

5.3.3 Analysis

Root Mean Square Error We use the commonly used root mean square error (RMSE) to measure the distance (in degrees) between the real and generated trajectories. We report the mean of pairwise RMSE between the real and the generated trajectories, and compare it with the previous gait generation methods.

$$
\begin{aligned}
\textit{mean pairwise RMSE} &= \frac{\sum_i \sum_j \sum_{n_{ij}} \sum_{m_{ij}} RMSE(x_{m_{ij}}, \hat{x}_{n_{ij}})}{\sum_i \sum_j \sum_{n_{ij}} \sum_{m_{ij}} 1} \\
&\text{i = person index, j = speed index} \\
m_{ij} &= \text{real trajectory index for person i and speed j} \\
n_{ij} &= \text{generated trajectory index for person i and speed j} \\
x_{m_{ij}} &= m_{ij}^{th} \text{ real trajectory for person i and speed j} \\
\hat{x}_{n_{ij}} &= n_{ij}^{th} \text{ generated trajectory for person i and speed j}
\end{aligned}
\tag{5.1}
$$

Person specificity and speed specificity As described in Section 5.2.2, we use classifiers to quantify the quality of generated trajectories. If the generated trajectories can fool classifiers trained on real trajectories, then the generator can be considered successful. The strongest example of this would be if the generator can fool classifiers from a different model class. Since we use neural network classifiers within the generator, (see Section 5.3.5) we first need to see how successful non-neural-network classifiers can be. These classifiers (see Table 5.2) were trained on all the ground truth data (both training and test sets), validated using 5-fold cross-validation. We chose a set of well-established classifier types whose behavior is well understood, and are arguably more interpretable and can be vetted by clinicians.

We trained the following 5 model types- Decision Trees, Linear Discriminant, Naive Bayes, Support Vector Machines (SVM) and K-Nearest Neighbor (KNN) using the MATLAB 2020b classification learner toolbox, to separately classify real trajectories in terms of person and speeds. The hyperparameters corresponding to the best model for each model type are listed in Table 5.1 The SVM models perform the best with person classification accuracy of 99.8% and speed classification accuracy of 83.9%. Table 5.2 shows the speed and person classification performance for the best model in each model type.

The second purpose of the classifiers is to guide training. *Neural Network* classifiers are used to regularize latent spaces and design cost functions, as described in section 5.3.5. Using Neural Networks classfiers allows differentiable layers through which gradients can be computed and backpropagated, and can be easily integrated with the Encoder-Decoder

Table 5.1: Hyperparameters for person and speed classifier

Models	**Person classifier**	**Speed classifier**
Decision Trees	Preset: Fine Tree; max splits = 100; Split criterion: Gini's diversity index	Preset: Fine Tree; max splits = 100; Split criterion: Gini's diversity index
Linear Discriminant	Covariance structure: Full	Covariance structure: Full
Naive Bayes	Preset: Kernel Naive Bayes; Distribution: Kernel smoothing; Kernel: Gaussian; Support: Unbounded	Preset: Kernel Naive Bayes; Distribution: Kernel smoothing; Kernel: Gaussian; Support: Unbounded
Support Vector Machine	Kernel: Quadratic; Kernel scale: Automatic; Box constraint level: 1; Multiclass method: One-vs-One; Standardize data: true	Kernel: Cubic; Kernel scale: Automatic; Box constraint level: 1; Multiclass method: One-vs-One; Standardize data: true
K Nearest Neighbors	Preset: Weighted KNN; Number of neighbors = 10; Distance metric: Euclidean; Distance weight: Squared inverse; Standardize data: true	Preset: Weighted KNN; Number of neighbors = 10; Distance metric: Euclidean; Distance weight: Squared inverse; Standardize data: true

Table 5.2: Person and speed classification performance on real trajectories

Accuracy	**Tree**	**LDA**	**Naive Bayes**	**SVM**	**KNN**
Person Classification	91.8 %	99.5 %	98.2 %	**99.8** %	99.5 %
Speed Classification	38.1 %	52.5 %	40.6 %	**83.9** %	76.6 %

backbone see Fig. 5.5. These classifiers are trained on only the real trajectories from the training data to prevent leakage from the test data. The classifiers are validated using 5-fold cross validation to ensure robustness in hyperparameter selection.

5.3.4 Baseline Encoder-Decoder Architecture

Based on the concept described in Fig. 5.3 and the supervised adversarial autoencoder used for hand writing style transfer on the MNIST dataset in [65], we use an encoder-decoder architecture. The encoder extracts the information independent of speed i.e. person-specific style. The decoder synthesizes the trajectories by combining the style information in 'z', with the desired speed information injected using one-hot encoding in 'y'. The model is trained by reconstructing the input trajectories with RMSE as the loss function. This is different from the MNIST style transfer in that the output for each dimension in the MNIST case is binary and hence the task can be framed as pixel-wise classification problem, while here the task is fundamentally a regression problem due to continuous trajectories. This model serves as our baseline for performance and we introduce regularization on top of this architecture in the next section to improve over the baseline.

Encoder details The architecture of the baseline encoder-decoder is shown in Fig. 5.5. The encoder has 4 layers, which transform R^{300} input trajectories to their R^3 latent style representation. Latent style space is chosen to be 3-D to make visualization easy. The first 2 layers of the encoder have *leaky-ReLU* activation with a slope of 0.1 for negative values. The following two layers use linear activation. We do not present an exhaustive search

of architecture choices here; the presented architecture generally resembles our previous successful learned latent representations for human movement [80, 11].

Decoder details The decoder (Fig. 5.5) takes the style representation of the input trajectory concatenated with one-hot representation of the corresponding speed, and uses it to reconstruct the input trajectory. The decoder has 4 layers, with the first 3 layers using *leaky-ReLU* activation functions with a slope of 0.1 for negative values. The final layer uses a linear activation function. A root-mean-squared reconstruction error loss is minimized to reduce the distance between the real and generated trajectories. (Eq. 5.2)

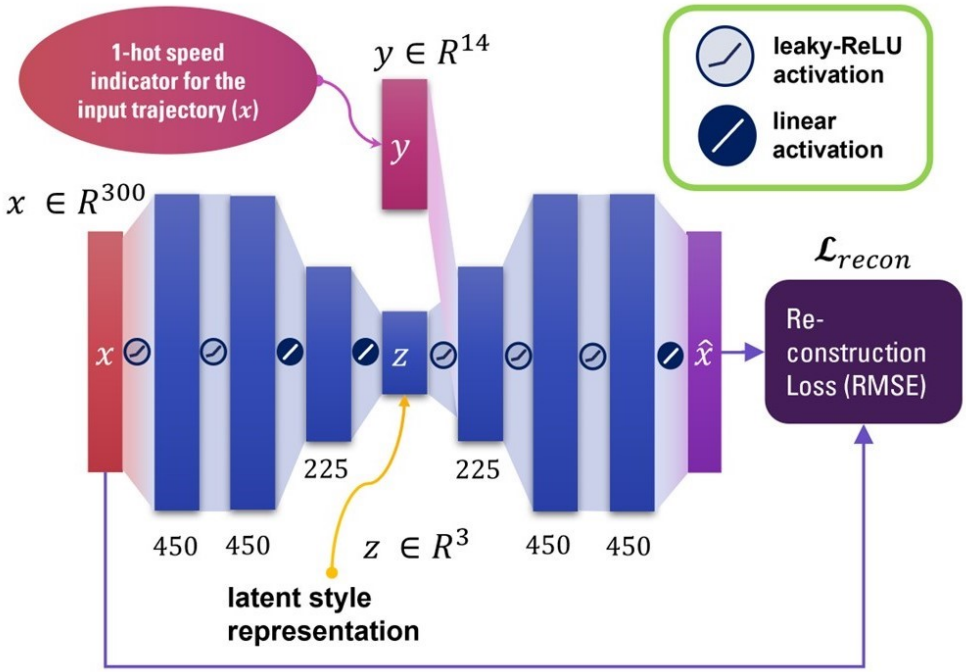

Figure 5.5: Baseline style transfer architecture, based on [65]. The encoder extracts person specific style from the input trajectories and the decoder uses the style information and desire speed concatenated at the latent space, to generate person and speed specific trajectory.

$$\mathcal{L}_{total} = \mathcal{L}_{recon} \tag{5.2}$$

5.3.5 Gait generation using latent space regularization and cost function design

The baseline encoder-decoder architecture uses speed injection at the decoder input to encourage the encoder to learn the complementary information i.e. personal style. This however is contingent on factors like the size of latent space 'z', and doesn't allow much control over what information is preserved in 'z'. This also doesn't allow us to control how the decoder combines 'y' and 'z' to generate the trajectory. Thus, we introduce two additional changes: 1) cost function design at the output of the decoder to emphasise the desired attributes in the synthesized trajectories, and 2) latent space regularization to control what information is encoded in the 'z' space.

Output regularization using cost function design We used person and speed classifiers, trained on real trajectories from the training set, to regularize the model at the output via terms in the loss function. The classifiers take the trajectories generated by the decoder as input, and output the person and speed classification losses, which are weighed by corresponding parameters α and β, and added to the reconstruction RMSE loss. Grid search over α and β is used to find the optimal weights. These classifiers enforce desirable traits i.e. person-specificity and speed-specificity in the outputs from the decoder. In this way, we are directly rewarding the system for achieving our goal - to creating synthetic trajectories that resemble the person or the speed, beyond RMSE. This approach is depicted in Fig. 5.6.

$$\mathcal{L}_{total} = (1 - \alpha - \beta) * \mathcal{L}_{recon} + \alpha * \mathcal{L}_{person} + \beta * \mathcal{L}_{speed} \tag{5.3}$$

Latent space regularization Using classifiers at the output doesn't allow us to explicitly control the attributes encoded in the latent space 'z'. The designer's control is only exerted at the cost function, but it's up to the learning algorithm to organize how the latent space

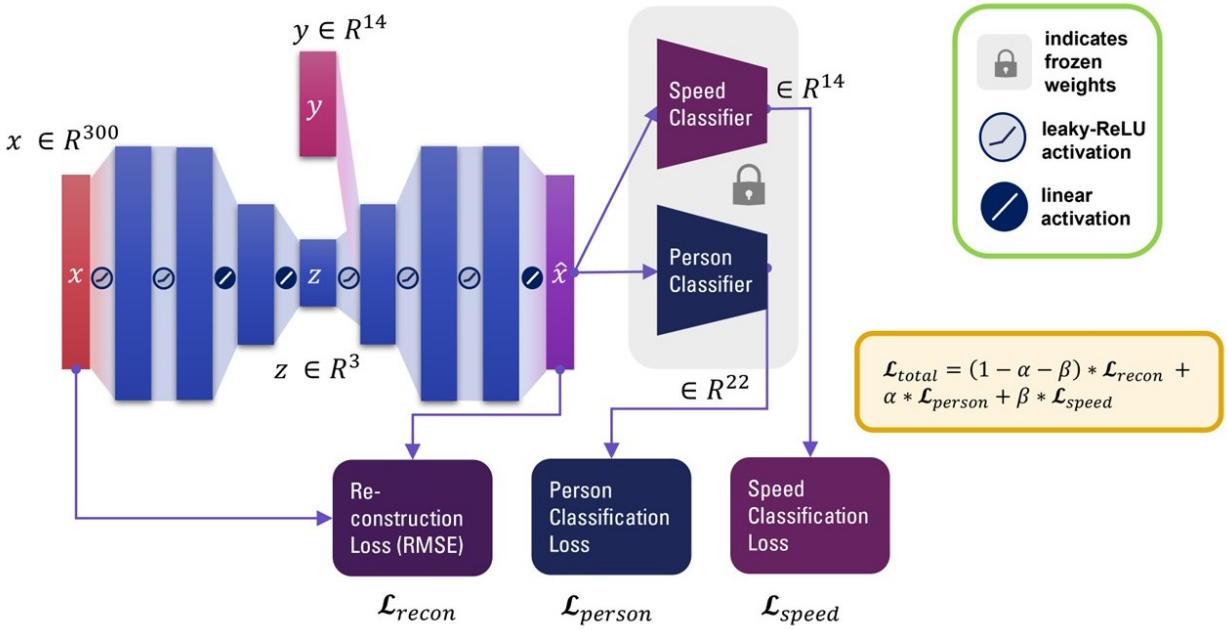

Figure 5.6: Output Regularization using cost function design. In addition to the reconstruction loss, classifier losses are added, weighted by coefficients $\alpha$and β.

is encoded and decoded. This can be difficult to interpret, and doesn't provide a detailed way to encourage distinct parts of the architecture to take on distinct purposes. To address this, we use latent space regularization to directly force the latent space to encode person-specific information, while penalizing the inclusion of speed-specific information (See Fig. 5.7). Latent regularization is done in two steps: First, the encoder is trained to take real trajectories as input and produce a person classification. A negative speed classification cost is added to remove speed information in 'z'. These cost functions are weighed by a parameter 'γ', which is tuned using grid search. In the second step, after the encoder has been completely trained, the decoder portion of the architecture is trained to minimize reconstruction loss, with the encoder weights frozen. This prevents encoder representations learned in step 1 from changing. This approach is depicted in Fig. 5.7.

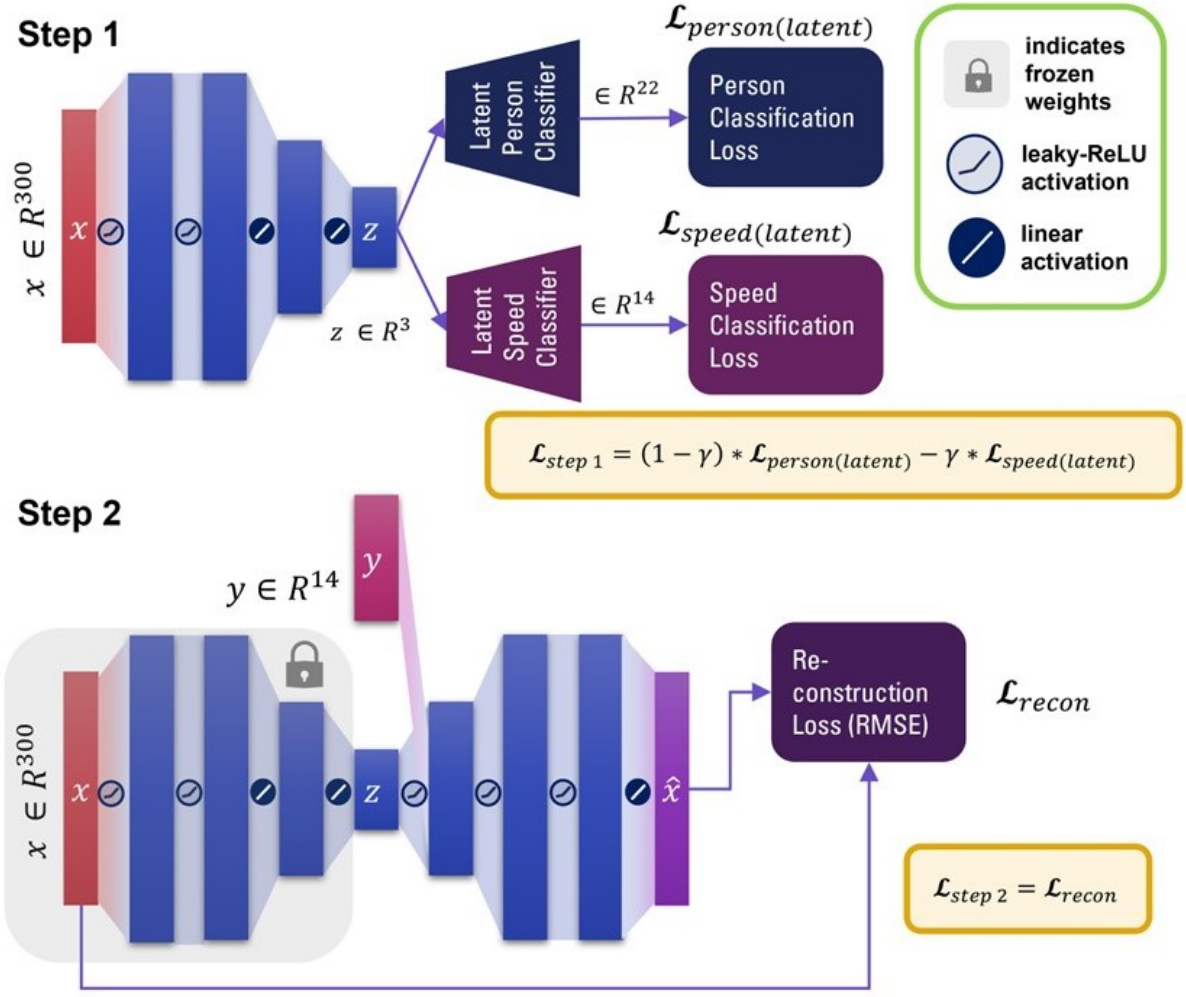

Figure 5.7: Latent Space Regularization. The encoder is trained first. Latent representations are regularized by classifiers in the latent space. The person classifier should be successful, indicating that latent representations of individuals are separated in the latent space. Speed classification, on the other hand, should fail, indicating that speed information is not encoded into the latent space. This is reflected by the negative coefficient for the speed loss term. The trained encoder weights are fixed, and the decoder is trained as in the baseline according to reconstruction loss.

$$\begin{aligned} \mathcal{L}_{step\ 1} &= (1-\gamma) * \mathcal{L}_{person(latent)} - \gamma * \mathcal{L}_{speed(latent)} \\ \mathcal{L}_{step\ 2} &= \mathcal{L}_{recon} \end{aligned} \tag{5.4}$$

Combining output and latent regularizations Output regularization helps us emphasize desired attributes in the output of the decoder, while latent space regularization gives us explicit control over attribute specific representations learned in the latent space 'z' and therefore what information is available to the decoder for trajectory synthesis. These methods have mutually exclusive advantages and combining them could lead to overall better performance. Thus, we present a model that combine the two methods. The combined model is trained by first pretraining the encoder similar to latent regularization (Fig.5.7; Step 1). Then, the entire model is trained like step 2 with person speed classification cost applied to the output of the decoder.

$$\begin{aligned} \mathcal{L}_{step\ 1} &= (1-\gamma) * \mathcal{L}_{person(latent)} - \gamma * \mathcal{L}_{speed(latent)} \\ \mathcal{L}_{step\ 2} &= (1-\alpha-\beta) * \mathcal{L}_{recon} + \alpha * \mathcal{L}_{person} + \beta * \mathcal{L}_{speed} \end{aligned} \tag{5.5}$$

5.3.6 Calibration using target user examples

Everyone has a unique walking *"style"* which depends on several factors like height, weight, medical history etc. Thus, we require a few sample trajectories from the target user, to extract style representations and calibrate our models to the target users.. These 'calibration samples' can be obtained by capturing target user gait at a few speeds. In the case of prostheses/orthoses users these trajectories can be obtained by handtuning as per the standard clinical practices i.e. tuned trajectories at selected slow, medium and fast speeds. Then, trajectories at rest of the speeds can be interpolated or extrapolated.

For comparison among baseline, output regularization, latent regularization, and combined, we used sample trajectories at 3 speeds: slow (0.7 m/s), medium (1.1 m/s), and fast (1.5 m/s), mixed with trajectories from a databank from subjects with no mobility impairments, for training. Three example speeds were chosen because it is a reasonable amount of

data to expect a user to provide during a single visit. The three speeds span from the slowest to the fastest speeds to capture the maximal variations due to speed. However, we wanted to determine how sensitive the results are to this choice. Therefore, we have conducted another experiment using fewer example speeds. The best method from the above comparison was chosen (combined, see Section 5.4) and trained using fewer speeds from the target user data training data (Fig. 5.8). The easiest mode for the system has the most examples trajectories, at 3 speeds (slow, medium, and fast) from the target user. The moderate mode has only 2 speeds (slow and medium), and the hardest mode has only 1 speed (slow), included in the training data.

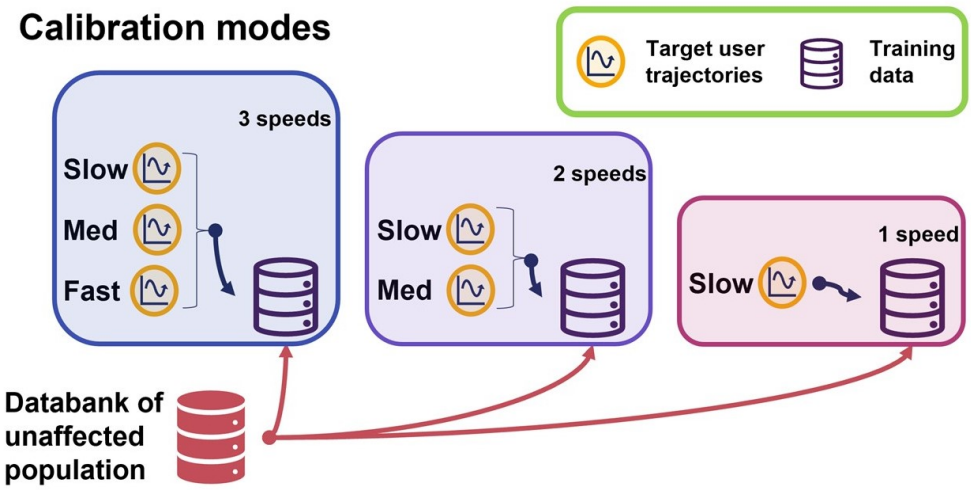

Figure 5.8: Calibration modes: Easy mode is where trajectories at 3 speeds from target users are included in the training data. Moderate mode includes 2 speeds while the hard mode only includes slow speed.

5.3.7 Hyperparameter Tuning

We compared 4 methods in this chapter, all using variations on the encoder-decoder architecture. The tuned hyperparameters were learning rates, person cost weight (α), speed cost

weight (β) and latent speed cost weight (γ). Grid search was used to find the optimal hyperparameters. A random selection of 20% samples from the training data is used as validation set for optimal hyperparameter selection (not used for model weights updates). Baseline and output regularization are trained in a single step. The range of hyperparameter values tested and the optimal values for both the methods are shown in Table 5.3. Latent regularization and combined regularization are trained in 2 steps. The first step trains the encoder and the second step trains the decoder. Table 5.4 shows the corresponding hyperparameters.

In the original work with hand writing style transfer, [65] regularization using an adversarial network was used as well, to force enforce a Gaussian distribution in the latent space. In our experiments we found that using adversarial network made training difficult and reduced the overall performance for the baseline model in terms of the key RMSE metric. Therefore, we set the learning rate for the adversarial update to 0 for all the experiments. Thus, the Gaussian distribution was not enforced on the latent space.

Table 5.3: Optimal hyperparameters for baseline, and output regularization

Hyperparameters	**search range**	**baseline**	**output reg.**
Learning rate	$[10^{-6} : 10^{-1} : 10^{-2}]$	10^{-5}	10^{-4}
Epochs	10000	-	-
Person cost weight (α)	[0.0 : 0.1 : 0.5]	n/a (0.0)	0.4
Speed cost weight (β)	[0.0 : 0.1 : 0.5]	n/a (0.0)	0.4

For step 1 of latent regularization, the use of negative weight for the speed classification loss sometimes led to exponential blowup of the speed cost term, effectively stagnating improvements in person loss. We found that very small cost weights were required to train the encoder effectively (Table 5.4). For the combined method, we used the encoder from step 1 of latent regularization.

Table 5.4: Optimal hyperparameters for latent regularization, and combined regularization

Hyperparameters	**search range**	**latent reg.**	**combined**
Encoder learning rate	$[10^{-5} : 10^{-1} : 10^{-3}]$	10^{-5}	10^{-5}
Encoder Epochs	10000	-	-
Decoder learning rate	$[10^{-5} : 10^{-1} : 10^{-2}]$	10^{-4}	10^{-4}
Encoder Epochs	5000	-	-
Latent speed cost weight (γ)	$[10^{-13} : 10^{-1} : 10^{-7}]$	10^{-10}	10^{-10}
Person cost weight (α)	$[0.0 : 0.1 : 0.5]$	n/a (0.0)	0.2
Speed cost weight (β)	$[0.0 : 0.1 : 0.5]$	n/a (0.0)	0.2

5.4 Results and Discussion

Table 5.5 shows the performance of different methods in terms of RMSE, person classification accuracy and speed classification accuracy. The leftmost column of Table 5.5 shows pairwise RMSE between two real trajectories of same speed and person. This indicates the inherent variability of the real trajectories, even for a given person and speed. The baseline model, which we adapted from handwriting style transfer [65], was capable of generating trajectories with RMSEs of 1.29, 2.87 and 1.78 degrees for hip, knee and ankle respectively. For comparison, the method shown in [91] show RMSE values of 2.31, 3.46 and 2.60 degrees respectively. Thus, the baseline model performs slightly better in terms of RMSE.

However, the trajectories generated by the baseline method show relatively low performance for person and speed appropriateness. Even though the RMSEs are similar (Table 5.5, first 3 rows), the classification success is quite poor (Table 5.5, bottom two rows.) The trajectories are not capturing the unique features that make them person and speed specific. This supports our initial guess that RMSE might not necessarily capture other attributes like personal style and speed. This also provides evidence that there is a need to use other metrics to assess our models, similar to how we use the SVM person and speed classifiers

here.

The baseline style transfer method also doesn't provide explicit control over attributes of generated trajectories. For example, if an application prioritized accurate person specificity, with less importance placed on speed, there is no straightforward way to adjust the method accordingly. As shown in Fig. 5.5, the decoder simply takes in the one-hot vector y representing the desired speed, and combines that with the latent representation of personal style, z, to output a trajectory that is penalized for reconstruction loss. In exploratory experiments, we varied the number of dimensions of the latent representation z. We found that a low number of dimensions would lead to better speed performance while reducing person performance, while a higher number of dimensions would result in better person performance and bad speed performance.

5.4.1 Regularizations

Neural network classifiers trained only on the training data were used to penalize speed and person performance, allowing us to emphasize these desired attributes in the generated trajectories, along with low RMSE, by designing a cost function including classifier loss terms. Table 5.5 shows that output regularization leads to performance improvement over the baseline in terms of speed and person classification as well as RMSE. While this enhances overall performance, it doesn't allow us to control the representations learned in the latent space 'z'. To do this, we regularized the latent space explicitly, which also results in overall performance improvement over the baseline. The performance improvement is drastic in terms of person specificity, but not as great for speed specificity. This is in contrast to output regularization, which shows drastic improvement in terms of speed-specificity.

By combining the two complementary approaches, we get a system that provides almost as successful person classification, and much improved speed classification (Table 5.5 Combined, rightmost column). So we conclude that for this data, the combined strategy is the overall best if one cares about both person and speed classification simultaneously.

Table 5.5: Comparison of the baseline and regularization methods. The leftmost column shows the mean of RMSE between each pair of real trajectory from the same class (person and speed), and the performance of person and speed classifiers on real trajectories (See Table 5.2).

Performance	**Real**	**Baseline**	**Output Reg.**	**Latent Reg.**	**Combined**
# target user samples	-	166	166	166	166
# generated trajectories	-	2324	2324	2324	2324
Hip mean RMSE (deg)	1.03	1.29	1.18	**1.08**	*1.12*
Knee mean RMSE (deg)	2.30	2.87	2.60	**2.45**	*2.54*
Ankle mean RMSE (deg)	1.48	1.78	1.63	**1.49**	*1.57*
Person Classification	99.8 %	81.67 %	89.37 %	**99.78** %	*99.01* %
Speed Classification	83.9 %	30.81 %	**80.16** %	49.31 %	*64.16* %

5.4.2 Latent Space Visualization

Using bottlenecked, low-dimensional, latent spaces is not only advantageous in terms of overall model performance, but also allows for visualization or manipulation of the encoded data in a tractable space. Fig. 5.9 shows the latent style representations learned by the baseline model (left) and the two regularization methods (center and right) from 2 viewpoints each. The latent space of the combined approach is the same as the latent regularization only, because the same encoders were used. Each subject is assigned a unique color. The orientation of axes was chosen so that the target users are clearly visible for each method. We see that output regularization doesn't greatly change the pattern and arrangement of subjects in the latent space from the baseline method or make the subjects more separable. It does, however, create better clusters for the target users, albeit not completely separable from the other users. Latent regularization distinctly makes tighter clusters, with easily separable subjects. It also tends to push subjects far apart from each other in the radial direction.

This indicates the usefulness of pretraining the attribute specific encoder (here style encoder) with emphasis on removing information related to other attributes (here speed). This creates more interpretable and unadulterated representations, which can then be used for various downstream tasks, e.g. personalization. In this chapter we used these representations for a specific task i.e. to generate gait trajectories at a few desired speeds. However, these representations can potentially be used to personalize other predictive models of gait [83, 85, 104], instead of using more standard quantifiable attributes like height, weight etc.

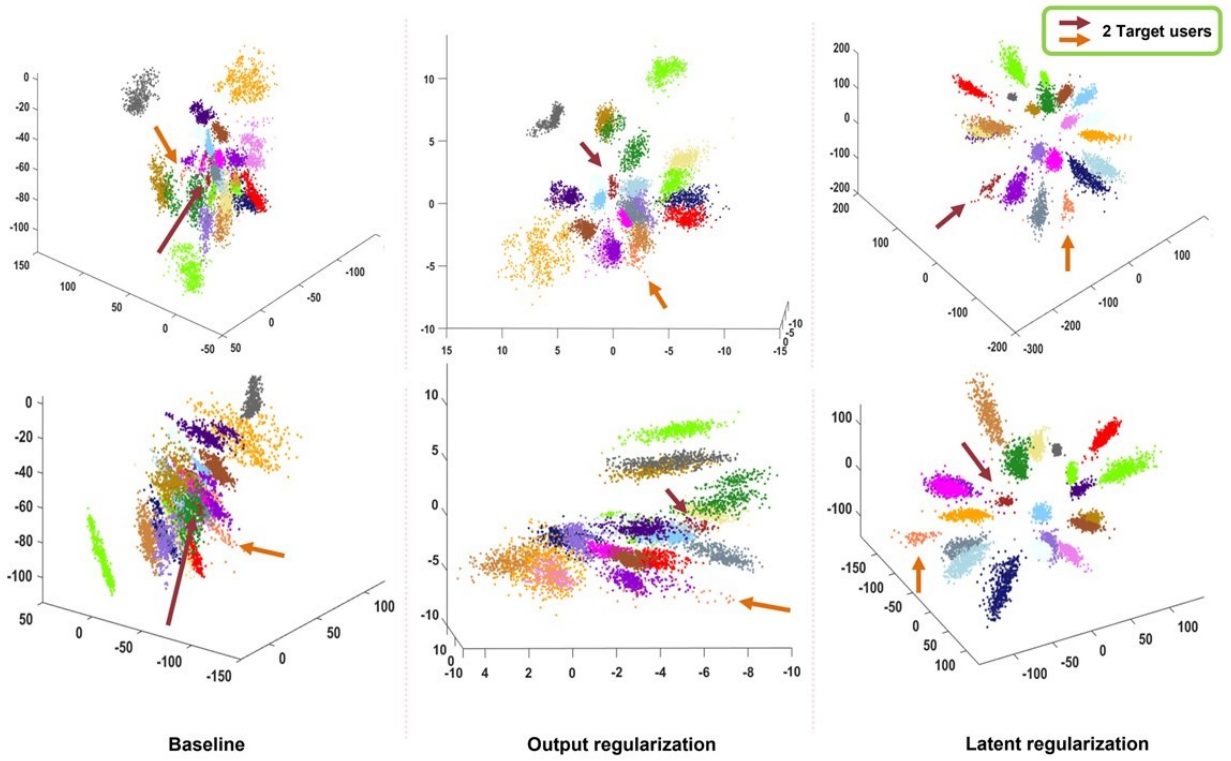

Figure 5.9: Latent space visualization: Two different views of latent encoding of the trajectories from all the subjects, for baseline, output regularization and latent regularization methods are shown here. The encoding for the combined regularization is same as that for latent regularization. Different colors correspond to different subjects The two arrows indicate the two target users.

5.4.3 Amount of calibration data

We performed a sensitivity analysis to understand how the results would degrade given fewer examples. We chose two additional relatively challenging scenarios: a moderate version where the user provides walking examples at 0.7 m/s and 1.1 m/s, and a hard version where the user provides walking examples only at 0.7 m/s. For all three, the combined latent and output regularization strategy was used. As can be seen in Table 5.6, performance degradation is quite small. We see a worst-case reduction of 1.03% in person classification accuracy and even a 0.13% increase in speed classification accuracy. Considering the variability of trajectories (See Table 5.5; leftmost column), this appears to be functionally equivalent. From this experiment we infer that a single example speed should be sufficient to use the proposed system. This is a significant result since this has the potential to greatly reduce the amount of effort required to tune prostheses controllers, which currently limits their capabilities.

Table 5.6: Performance for the sensitivity analysis to the number of example speeds given.

	3 speeds	**2 speeds**	**1 speed**
# target user samples	166	102	46
# generated trajectories	2324	1428	644
Hip mean RMSE (deg)	1.12	1.12	1.13
Knee mean RMSE (deg)	2.54	2.54	2.56
Ankle mean RMSE (deg)	1.57	1.58	1.59
Person Classification	99.01 %	98.74 %	97.98 %
Speed Classification	64.16 %	64.35 %	64.29 %

5.4.4 Beyond Personalization

In this chapter we showed an application of the concept illustrated in Fig. 5.3, where we chose personalization of generated gait at different speeds. However, the methods are application

agnostic. With the recent rise in multimodal gait datasets [100, 17, 105, 76, 97, 63, 102], we can also learn representations of hard to define factors in a similar way the representations of style were learned in this chapter. Similarly, we could encode terrain in an appropriate latent space and then compose different representations using the decoder. As we have shown here, these approaches to representation learning are compatible with optimization of the decoder by including attribute specific cost terms, and these can be related to the performance of classifiers. This means that we don't necessarily have to have first-principles understanding of what makes the classes different, if we can simply sample many examples from them.

5.4.5 Limitations

This work does not present any analysis of the clinical effects or advantages of personalized prosthesis control. Whether a more personalized trajectory would lead to any improvements in the quality of life of the target users stands to be tested and should be investigated in future works.

The total amount of data available for learning models was relatively small, 9073 trajectories, compared to millions of samples that are present in some datasets. Even [65], upon which the baseline model was based, used the MNIST dataset which is comprised of over 60,000 samples. Thus, it remains to be tested how cost function design compares with latent space regularization when using dramatically larger datasets. Regardless, for small datasets this chapter shows the strengths and weakness of the different methods.

In this work, we represented 14 speeds using 14-dimensional 1-hot encodings, similar to how [65] used digits, but speed being a continuous signal arguably can be best represented a single continuous variable. The effect of this choice can be invested in the future work.

Using more interpretable classifiers for quantification of attribute-specific performance (speed or style) is ideal for communicating the trustworthiness of the generated trajectories to stakeholders. In this work, we used SVMs with quadratic and cubic kernels for person and speed classifiers, since they yielded the best classification performance for real trajectories. However, more interpretable methods like KNN [71], which perform good enough (see Table

5.2) can be used as well.

In this chapter we don't present an exhaustive analysis of how the cost function weights α, β and γ influence the performance of the models. We only presented the best performance achieved by each model in a sparse grid search. However, understanding the behaviour of model under interaction of different cost terms can be useful if we want to downplay or emphasize an attribute over the other.

5.4.6 Future Work

A study with users using a device controlled to achieve these trajectories would be necessary to understand the effects of personalization. We have limited the scope of the chapter to generating the trajectories, but what trajectory variations are important, and what properties result in clinical meaningfulness, remains to be shown.

We used person classification along with negative cost on speed classification to regularize the latent space. This is a variation on contrastive learning [54], which is a deep and active topic in modern representation learning. In contrastive learning, samples from different classes are explicitly pushed apart from each other in the latent space, not necessarily by using classifier loss. Examples from the same classes can be pushed closer to one another as well. This is similar in spirit to what we present here and would likely have different tradeoffs that would be potentially fruitful to explore.

We do not present a rigorous search over the architecture size and type. However, we anticipate that convolutional neural networks (CNN) or recurrent neural networks (RNN) instead of the fully connected network could further improve trajectory generation performance. This is because the real trajectories have a local temporal continuity built into them, and the the fully connected network generates each dimension (out of the 300 dimensions here) independently. CNN or RNN based encoders and decoders could emphasize this time-continuous aspect of the trajectories.

Chapter 6

CONCLUSION

The goal of this research is to make improvements in assistive device control and human movement analysis, using recent advances in wearable sensors, and data-driven modeling and analysis. To this end, we collected a novel dataset **(44 participants)** of 24 hours of human movement in out-of-the-lab environments, with the corresponding egocentric vision data for more challenging environments. The dataset is described in [102] and available on the figshare repositories Part 1 and Part 2. We then used the said dataset in following ways:

In chapter 2, we used deep neural networks trained on wearable motion capture data to build predictive models of human gait, with the objective of using the predictions as reference trajectories for assistive devices. We tested the effectiveness of the model at predicting several different activities, without explicit use of handcrafted features or segmentation of gait into phases like stance or swing. The prediction performance of our model in terms RMS error was comparable with other approaches that do continuous prediction of joint trajectories. However, our model also predicted CHAMP activities, which other approaches don't predict. We also showed that performance degraded as we increased the complexity of the modeled activity, thereby indicating greater need of data for more complex activities. In addition, we showed the predictive capabilities of the model in often unmodeled events in gait literature like transitions between activities like sidestepping to backward walking or a 180 deg turn. This was possible because wearable motion capture allows sampling a variety of such transitions, and thus models can learn to predict them. Other existing approaches have not yet demonstrated these capabilities partly because most existing gait datasets do not have examples of such events, but also because these methods usually exploit idiosyncrasies of the activity they are modeling, and thus can't easily be scaled to other activities.

In chapter 3, we included environment modeling in the predictive model of the gait, using egocentric vision. We showed that augmenting imu-based motion capture data with the corresponding first-person vision data improves gait prediction performance. The performance gain is greater in environments where obstacles are more densely packed and thus force the user to rely on their visual field. Thus, the visual data is more correlated with subject's subsequent behavior. We also presented scenarios where vision lead to degradation of performance thus demonstrating the need for a method to determine whether to trust vision or not.

In chapter 4, we analyzed the complexity of different activities captured using wearable motion capture system, according to several measures used in the gait literature, which measure different aspects of the data. We showed that using complexity as a unitary concept is not feasible as different measure tend to disagree about the relative complexity of different activities. We also found several commonly used measures to be contradictory with the human intuition about relative complexity of the activities. For example, dimensionality measured using PCA, ranks sidestepping to be of greater complexity than navigating in classrooms and Atrium, and backward walking is almost twice as complex as other activities. Finally, we presented some practical recommendations about when a measure can be considered reliable.

In chapter 5, we presented a general framework for generating gait, while being able to explicitly control the attributes (like style or speed), expressed in the generated gait. We demonstrated the effectiveness of the method on generating person-specific gait for several different speeds. In addition, we proposed using classifiers trained on all the data as adjudicators of performance of the generative models, particularly for attributes like personal style which can be hard to quantify. These classifiers, were of relatively lower complexity than deep neural networks and can considered more trustworthy than deep neural networks.

Thus, the presented research tackles some of the challenges posed by limited tools and technology in the past, to control of assistive devices and human gait analysis. We presented a novel out-of-the lab dataset, an expressive method for gait modeling (Aim 1), studied

the effectiveness of environment modeling in improving predictive models of gait (Aim 2), analyzed the complexity of locomotion activities and demonstrated the drawbacks in using some common measures of complexity in gait literature and proposed guidelines for their usage (Aim 3). and finally we presented a framework for handling interpersonal differences in control of assistive devices (Aim 4). This should open new avenues of research in gait analysis and assistive device control, and hopefully lead to fundamental changes in device capabilities.

Future work

In this work, we are only beginning to address the complexity of human gait, and some of its causes. It's evidently clear that a lot more work needs to be done before we can gain a clear understanding of how humans move and interact with their surroundings, and are able to create assistive devices that can restore the true complexity of unaffected human gait. In this section, we list a few research directions, associated with each of the 4 limbs of this dissertation, that can be investigated in the future:

Coordinated Movement

- The biggest hurdle in adoption of this approach for control of assistive devices, is the lack of trustworthiness associated with models' predictions. While the predictive model presented in this work can easily scale to include any desired activity unlike the existing approaches, it also lack the reliability associated with those approaches. Since, from the users' perspective having a smoother experience while using the device is paramount, the coordinate movement control approach needs to be made more reliable. One way could be to incorporate uncertainties associated with the model's prediction [52], in the controller. Safety mechanisms need to be built in to account for the cases when model predictions go wrong.

- Another challenge with the coordinated movement approach is that certain events can

be extremely rare in the dataset, e.g. to start walking from a complete stop, and thus training models without prioritizing such events can to poor performance for such events. Thus, approaches that can identify such critical events and weigh them appropriately in the loss functions, need to be explored.

- In our work, we only explored a simple root mean squared (RMSE) loss function for training our model, which tends to weigh all moments in time to be of equal importance. However, from the users perspective certain moments are more important than the others e.g. stance phase when prosthetic leg is on the ground is more critical than swing phase, and thus needs to be more accurately predicted. Thus, a loss function that weighs the stance phase more should be explored. This should also be relatively straightforward since our dataset also has foot contact information available. Similarly, relative weights of ankle and knee terms in the loss function should be investigated.

- Predicting the movements of an able-bodied population is only the first step toward using those predictions to create intelligent prosthesis control. Movements generated by able-bodied users almost certainly differ from those generated by prosthesis users. This gets at the heart of a central problem in rehabilitation training. What is the right gait that we should strive for? Should individuals undergoing rehabilitation be encouraged to walk exactly like able-bodied individuals, or since the prosthesis and amputation affect dynamics, should the gait be adapted to maximise the user's potential? In the former case significant training at the user end must be designed so that they can explore the behavior of the prosthesis and train themselves to use it. In the latter case models based on examples from able-bodied individuals could still serve as a seed for further optimization of generated trajectories.

- Finally, the true potential of coordinate movement approach to control assistive devices can only be achieved with a much larger amount of data than is currently available. As wearable motion capture systems gain maturity, they have the potential to do in

the domain of human movement analysis, and assistive devices, what internet did in the domain of text image and videos. However, collection of large amounts of clean data is still a major hurdle. The data collection process is extremely inefficient. Not only is donning the mocap suit and the calibration process time consuming, but the mocap system requires frequent monitoring and recalibration to prevent data corruption due to sensor drift. The calibration process also needs to be done in presence of an experimenter since it's not very convenient for the subject to do calibration by themselves. In addition, a relatively heavy device like laptop is required to monitor the data quality. Thus, the experimenter has to walk with the subject, while carrying the device. This could affect the natural behavior of the subject. Thus, applications to monitor and visualize the data on smaller devices like a mobile phone (e.g. MVN remote [4]) need to be improved, as well as calibration protocols that are easier for the subject to perform on their own, need to be developed. This should make the data collection process easier, and enable data collection in much larger quantities.

Environment modeling for gait prediction using egocentric vision

- The scope of this investigation was limited to determining if continuous estimation of gait can be improved using egocentric vision data, and how the prediction performance changes with different environments. In this study, we only leveraged motion information from vision. Studying how appearance information affects performance and its comparison with motion information could be useful in designing a prosthetic controller using vision.

- We processed vision and kinematics features separately and combined them using late fusion. Early mixing of the two modalities could be beneficial towards prediction. Particularly, explicitly combining vision frames along with corresponding neck and torso orientations could be useful.

- Improving performance overall is desirable, but to translate these findings toward practical use it might be necessary to create a method for deciding when vision is appropriate, i.e. models that provide uncertainty estimates [52] and other Bayesian style approaches [9].

- In this work, we normalized kinematics data using max-min scaling as proposed in [83]. This scaling method could be non-optimal as outliers can determine the scale of the data. This is particularly problematic when sensors drop causing larger unnatural deviations in joint angles. We manually filtered trials with such events. However, a potentially better approach would be to scale the data using mean and standard deviation

- The results we presented in this study are for performance of the predictors on a subject that was completely held out of the training set. They are representative of how this approach would perform on an unknown user with no prior training examples. This is a challenging version of the application, because the system must generalize to user-specific gait style, behaviors like head movements and other tics, as well as to an uncontrolled changing environment. Tuning the system to a new user using an approach similar to that highlighted in [124] or by including some brief walking data from the user in the training set, might help performance. In addition, the body dimensions of the user, (especially height) also influence where the eye-tracker is placed. Thus, fusing body dimensions or other measurements with the visual stream might further help performance, as in [34].

- The dataset collected as a part of this study also has gaze data from the user. From data visualization it was clear that gaze of the person contains information about their future path particularly while moving in congested spaces (see video). The effect of gaze on gait prediction should be investigated in the future.

Complexity analysis of locomotion activities

- One surprising finding of this study was that backward walking and sidestepping showed greater complexity than even classroom and atrium according to a few measures, particularly for the gait cycle variability measure- *gaitSD*, and the gait signal irregularity measure- *multi-scale entropy*. This can potentially be attributed to these activities being less practised than navigating around obstacles, or could be a drawback of the gait complexity measure used. Which one is the true cause needs to be determined, and whether practice plays a role in gait complexity, should be investigated.

Optimizing representations of multiple simultaneous attributes for personalized gait generation using deep learning

- In this study, we only generated gait for 2 attributes i.e. style and speed. However, it is possible to extend our approach to generate gait any desired set of attributes like style, mood, speed, incline, terrain etc. (See Fig. 5.3) This would require datasets that emphasize these attributes. With improving motion capture systems, such datasets are expected to proliferate, and should enable generating gait for any clinician specified measure of success.

- A study with users using a device controlled to achieve these trajectories would be necessary to understand the effects of personalization. We have limited the scope of this project to generating the trajectories, but what trajectory variations are important, and what properties result in clinical meaningfulness, remains to be shown.

- We used person classification along with negative cost on speed classification to regularize the latent space. This is a variation on contrastive learning [54], which is a deep and active topic in modern representation learning. In contrastive learning, samples from different classes are explicitly pushed apart from each other in the latent space, not necessarily by using classifier loss. Examples from the same classes can be pushed

closer to one another as well. This is similar in spirit to what we present here and would likely have different trade-offs that would be potentially fruitful to explore.

- We did not perform a rigorous search over the architecture size and type. However, we anticipate that convolutional neural networks (CNN) or recurrent neural networks (RNN) instead of the fully connected network could further improve trajectory generation performance. This is because the real trajectories have a local temporal continuity built into them, and the the fully connected network generates each dimension (out of the 300 dimensions here) independently. CNN or RNN based encoders and decoders could emphasize this time-continuous aspect of the trajectories.

BIBLIOGRAPHY

[1] Carnegie Mellon University - CMU Graphics Lab - motion capture library.

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

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

[4] MVN Remote for ios and android.

[5] Mvn user manual. https://www.xsens.com/software-downloads. (Accessed on 12/28/2021).

[6] Announcing project aria: a research project on the future of wearable ar. "https://tech.fb.com/announcing-project-aria-a-research-project-on-the-future-of-wearable-ar/", 2020. Accessed on 10/15/21.

[7] Luís Amaral, Plamen Ch Ivanov, Ary Goldberger, Michael G Rosenblum, Shlomo Havlin, H Eugene Stanley, and Zbigniew Struzik. Multifractality in human heartbeat dynamics. In *APS March Meeting Abstracts*, pages W12–02, 1998.

[8] 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.

[9] Max Berniker and Konrad Kording. Bayesian approaches to sensory integration for motor control. *Wiley Interdisciplinary Reviews: Cognitive Science*, 2(4):419–428, 2011.

[10] Maria Cristina Bisi, Paola Tamburini, Giulia Pacini Panebianco, and Rita Stagni. Nonlinear Analysis of Human Movement Dynamics Offers New Insights in the Development of Motor Control During Childhood. *Journal of Biomechanical Engineering*, 140(11), 08 2018. 111002.

[11] David Boe, Alexandra A Portnova-Fahreeva, Abhishek Sharma, Vijeth Rai, Astrini Sie, Pornthep Preechayasomboon, and Eric Rombokas. Dimensionality reduction of human gait for prosthetic control. *Frontiers in Bioengineering and Biotechnology*, page 925, 2021.

[12] Justin A Brantley, Trieu Phat Luu, Sho Nakagome, Fangshi Zhu, and Jose L Contreras-Vidal. Full body mobile brain-body imaging data during unconstrained locomotion on stairs, ramps, and level ground. *Scientific data*, 5(1):1–10, 2018.

[13] J Burdack, F Horst, S Giesselbach, I Hassan, S Daffner, and WI Schöllhorn. A public dataset of overground walking kinetics in healthy adult individuals on different sessions within one day. *Mendeley Data*, 1(2), 2020.

[14] Michael A. Busa and Richard E.A. van Emmerik. Multiscale entropy: A tool for understanding the complexity of postural control. *Journal of Sport and Health Science*, 5(1):44–51, 2016.

[15] D. J. Butler, J. Wulff, G. B. Stanley, and M. J. Black. A naturalistic open source movie for optical flow evaluation. In A. Fitzgibbon et al. (Eds.), editor, *European Conf. on Computer Vision (ECCV)*, Part IV, LNCS 7577, pages 611–625. Springer-Verlag, October 2012.

[16] Ugo H Buzzi, Nicholas Stergiou, Max J Kurz, Patricia A Hageman, and Jack Heidel. Nonlinear dynamics indicates aging affects variability during gait. *Clinical biomechanics*, 18(5):435–443, 2003.

[17] Jonathan Camargo, Aditya Ramanathan, Will Flanagan, and Aaron Young. A comprehensive, open-source dataset of lower limb biomechanics in multiple conditions of stairs, ramps, and level-ground ambulation and transitions. *Journal of Biomechanics*, 119:110320, 2021.

[18] Alejandro Cartas, Petia Radeva, and Mariella Dimiccoli. Activities of daily living monitoring via a wearable camera: Toward real-world applications. *IEEE Access*, 8:77344–77363, 2020.

[19] Guilhem Chéron, Ivan Laptev, and Cordelia Schmid. P-cnn: Pose-based cnn features for action recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 3218–3226, 2015.

[20] En-Fan Chou, Michelle Khine, Thurmon Lockhart, and Rahul Soangra. Effects of ecg data length on heart rate variability among young healthy adults. *Sensors*, 21(18), 2021.

[21] Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[22] Madalena Costa, Ary L Goldberger, and C-K Peng. Multiscale entropy analysis of biological signals. *Physical review E*, 71(2):021906, 2005.

[23] Leslie M Decker, Fabien Cignetti, and Nicholas Stergiou. Complexity and human gait. *Revista Andaluza de Medicina del Deporte*, 3(1):2–12, 2010.

[24] H Dimitrov, AMJ Bull, and D Farina. Real-time interface algorithm for ankle kinematics and stiffness from electromyographic signals. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 28(6):1416–1427, 2020.

[25] Jacques Duysens, Friedl De Groote, and Ilse Jonkers. The flexion synergy, mother of all synergies and father of new models of gait. *Frontiers in computational neuroscience*, 7:14, 2013.

[26] Kiana Ehsani, Hessam Bagherinezhad, Joseph Redmon, Roozbeh Mottaghi, and Ali Farhadi. Who let the dogs out? modeling dog behavior from visual data. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[27] Kiana Ehsani, Daniel Gordon, Thomas Hai Dang Nguyen, Roozbeh Mottaghi, and Ali Farhadi. What can you learn from your muscles? learning visual representation from human interactions. In *Proceedings of the International Conference on Learning Representations (ICLR)*, May 2021.

[28] Kyle R. Embry and Robert D. Gregg. Analysis of continuously varying kinematics for prosthetic leg control applications. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 29:262–272, 2021.

[29] Kyle R. Embry, Dario J. Villarreal, Rebecca L. Macaluso, and Robert D. Gregg. Modeling the kinematics of human locomotion over continuously varying speeds and inclines. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(12):2342–2350, 2018.

[30] 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.

[31] Mahdy Eslamy, Felix Oswald, and Arndt F Schilling. Estimation of knee angles based on thigh motion: A functional approach and implications for high-level controlling of active prosthetic knees. *IEEE Control Systems Magazine*, 40(3):49–61, 2020.

[32] Samuel Farmer, Barbara Silver-Thorn, Philip Voglewede, and Scott A Beardsley. Within-socket myoelectric prediction of continuous ankle kinematics for control of a powered transtibial prosthesis. *Journal of neural engineering*, 11(5):056027, 2014.

[33] Alireza Fathi, Ali Farhadi, and James M Rehg. Understanding egocentric activities. In *2011 international conference on computer vision*, pages 407–414. IEEE, 2011.

[34] 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.

[35] Andrew M Fraser and Harry L Swinney. Independent coordinates for strange attractors from mutual information. *Physical review A*, 33(2):1134, 1986.

[36] Kenzie B Friesen, Zhaotong Zhang, Patrick G Monaghan, Gretchen D Oliver, and Jaimie A Roper. All eyes on you: how researcher presence changes the way you walk. *Scientific Reports*, 10(1):1–8, 2020.

[37] Claudiane A Fukuchi, Reginaldo K Fukuchi, and Marcos Duarte. A public dataset of overground and treadmill walking kinematics and kinetics in healthy individuals. *PeerJ*, 6:e4640, 2018.

[38] Reginaldo K Fukuchi, Claudiane A Fukuchi, and Marcos Duarte. A public dataset of running biomechanics and the effects of running speed on lower extremity kinematics and kinetics. *PeerJ*, 5:e3298, 2017.

[39] 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.

[40] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[41] Georgia Gkioxari and Jitendra Malik. Finding action tubes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 759–768, 2015.

[42] Ary L Goldberger, Luis AN Amaral, Jeffrey M Hausdorff, Plamen Ch Ivanov, C-K Peng, and H Eugene Stanley. Fractal dynamics in physiology: alterations with disease

and aging. *Proceedings of the national academy of sciences*, 99(suppl 1):2466–2472, 2002.

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

[44] F Horst, S Lapuschkin, W Samek, KR Müller, and WI Schöllhorn. A public dataset of overground walking kinetics and full-body kinematics in healthy individuals. *Mendeley Data*, 2019.

[45] Yongshan Huang, Honglei An, Hongxu Ma, and Qing Wei. Modeling and individualizing continuous joint kinematics using gaussian process enhanced fourier series. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 2022.

[46] Anne Humeau-Heurtier. The multiscale entropy algorithm and its variants: A review. *Entropy*, 17(5):3110–3123, 2015.

[47] Junhwa Hur and Stefan Roth. Optical flow estimation in the deep learning age. In *Modelling Human Motion*, pages 119–140. Springer, 2020.

[48] Victoria Smith Hussain, Mark L. Spano, and Thurmon E. Lockhart. Effect of data length on time delay and embedding dimension for calculating the lyapunov exponent in walking. *Journal of The Royal Society Interface*, 17(168):20200311, 2020.

[49] MD Jacquelin Perry. Gait analysis: normal and pathological function. *New Jersey: SLACK*, 2010.

[50] M Hassan Jahanandish, Kaitlin G Rabe, Abhishek Srinivas, Nicholas P Fey, and Kenneth Hoyt. Task-invariant learning of continuous joint kinematics during steady-state and transient ambulation using ultrasound sensing. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 10536–10542. IEEE, 2021.

[51] 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.

[52] Alex Kendall and Yarin Gal. What uncertainties do we need in bayesian deep learning for computer vision? In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 5580–5590, Red Hook, NY, USA, 2017. Curran Associates Inc.

[53] Matthew B Kennel, Reggie Brown, and Henry DI Abarbanel. Determining embedding dimension for phase-space reconstruction using a geometrical construction. *Physical review A*, 45(6):3403, 1992.

[54] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33:18661–18673, 2020.

[55] Sotiris Kotsiantis, Dimitris Kanellopoulos, and Panayiotis Pintelas. Handling imbalanced datasets: A review, gests international transactions on computer science and engineering 30 (2006) 25–36. *Synthetic Oversampling of Instances Using Clustering.*

[56] Nili E. Krausz and Levi J. Hargrove. Sensor fusion of vision, kinetics and kinematics for forward prediction during walking with a transfemoral prosthesis. *IEEE Transactions on Medical Robotics and Bionics*, pages 1–1, 2021.

[57] Nili E. Krausz, Blair H. Hu, and Levi J. Hargrove. Subject- and environment-based sensor variability for wearable lower-limb assistive devices. *Sensors*, 19(22), 2019.

[58] B. Laschowski, W. McNally, A. Wong, and J. McPhee. Comparative analysis of environment recognition systems for control of lower-limb exoskeletons and prostheses. In *2020 8th IEEE RAS/EMBS International Conference for Biomedical Robotics and Biomechatronics (BioRob)*, pages 581–586, 2020.

[59] 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.

[60] Brokoslaw Laschowski, William McNally, Alexander Wong, and John McPhee. Environment classification for robotic leg prostheses and exoskeletons using deep convolutional neural networks. *bioRxiv*, 2021.

[61] Mark L Latash, John P Scholz, and Gregor Schöner. Toward a new theory of motor synergies. *Motor control*, 11(3):276–308, 2007.

[62] 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.

[63] Viktor Losing and Martina Hasenjäger. A multi-modal gait database of natural everyday-walk in an urban environment. *Scientific Data*, 9(1):1–12, 2022.

[64] Minghuang Ma, Haoqi Fan, and Kris M Kitani. Going deeper into first-person activity recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1894–1903, 2016.

[65] Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, and Ian J. Goodfellow. Adversarial autoencoders. *CoRR*, abs/1511.05644, 2015.

[66] Ernesto C Martinez-Villalpando, Luke Mooney, Grant Elliott, and Hugh Herr. Antagonistic active knee prosthesis. a metabolic cost of walking comparison with a variable-damping prosthetic knee. In *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pages 8519–8522. IEEE, 2011.

[67] 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.

[68] 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.

[69] Moritz Menze, Christian Heipke, and Andreas Geiger. Joint 3d estimation of vehicles and scene flow. In *ISPRS Workshop on Image Sequence Analysis (ISA)*, 2015.

[70] R. Meyes, Melanie Lu, C. W. D. Puiseau, and T. Meisen. Ablation studies in artificial neural networks. *ArXiv*, abs/1901.08644, 2019.

[71] Christoph Molnar. *Interpretable machine learning*. Lulu. com, 2020.

[72] Jason K Moore, Sandra K Hnat, and Antonie J van den Bogert. An elaborate data set on human gait and the effect of mechanical perturbations. *PeerJ*, 3:e918, 2015.

[73] Steven Morrison and Karl M Newell. Dimension and complexity in human movement and posture. *Nonlinear Dynamics, Psychology, and Life Sciences*, 2015.

[74] UG Myn, MVN Link, and MVN Awinda. Xsens mvn user manual, 2015.

[75] Lyle Noakes. The takens embedding theorem. *International Journal of Bifurcation and Chaos*, 1(04):867–872, 1991.

[76] Manuel Palermo, João M Lopes, João André, Ana C Matias, João Cerqueira, and Cristina P Santos. A multi-camera and multimodal dataset for posture and gait analysis. *Scientific Data*, 9(1):1–11, 2022.

[77] Andrea Parri, Elena Martini, Joost Geeroms, Louis Flynn, Guido Pasquini, Simona Crea, Raffaele Molino Lova, Dirk Lefeber, Roman Kamnik, Marko Munih, et al. Whole body awareness for controlling a robotic transfemoral prosthesis. *Frontiers in neurorobotics*, 11:25, 2017.

[78] C-K Peng, J Mietus, JM Hausdorff, Shlomo Havlin, H Eugene Stanley, and Ary L Goldberger. Long-range anticorrelations and non-gaussian behavior of the heartbeat. *Physical review letters*, 70(9):1343, 1993.

[79] Hamed Pirsiavash and Deva Ramanan. Detecting activities of daily living in first-person camera views. In *2012 IEEE conference on computer vision and pattern recognition*, pages 2847–2854. IEEE, 2012.

[80] Alexandra A Portnova-Fahreeva, Fabio Rizzoglio, Ilana Nisky, Maura Casadio, Ferdinando A Mussa-Ivaldi, and Eric Rombokas. Linear and non-linear dimensionality-reduction techniques on full hand kinematics. *Frontiers in bioengineering and biotechnology*, 8:429, 2020.

[81] Nannan Qin, Xiangyun Hu, and Hengming Dai. Deep fusion of multi-view and multimodal representation of als point cloud for 3d terrain scene recognition. *ISPRS Journal of Photogrammetry and Remote Sensing*, 143:205–212, 2018. ISPRS Journal of Photogrammetry and Remote Sensing Theme Issue "Point Cloud Processing".

[82] David Quintero, Dario J. Villarreal, Daniel J. Lambert, Susan Kapp, and Robert 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.

[83] 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.

[84] 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.

[85] Vijeth Rai, Abhishek Sharma, David Boe, Pornthep Preechayasomboon, and Eric Rombokas. Continuous and unified modelling of joint kinematics for multiple activities. *IEEE Access*, 2022.

[86] Vijeth Rai, Abhishek Sharma, Pornthep Preechayasomboon, and Eric Rombokas. Coordinated movement for prosthesis reference trajectory generation: Temporal factors and attention. In *2020 8th IEEE RAS/EMBS International Conference for Biomedical Robotics and Biomechatronics (BioRob)*, pages 939–945, 2020.

[87] 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.

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

[89] 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.

[90] Emma Reznick, Kyle Embry, and Robert D. Gregg. Predicting individualized joint kinematics over a continuous range of slopes and speeds. In *2020 8th IEEE RAS/EMBS International Conference for Biomedical Robotics and Biomechatronics (BioRob)*, pages 666–672, 2020.

[91] Emma Reznick, Kyle Embry, and Robert D. Gregg. Predicting individualized joint kinematics over a continuous range of slopes and speeds. In *2020 8th IEEE RAS/EMBS International Conference for Biomedical Robotics and Biomechatronics (BioRob)*, pages 666–672, 2020.

[92] Emma Reznick, Cara Welker, and Robert Gregg. Predicting individualized joint kinematics over continuous variations of walking, running, and stair climbing. *TechRxiv*, 2022.

[93] Eric Rombokas, Mark Malhotra, Evangelos A Theodorou, Emo Todorov, and Yoky Matsuoka. Reinforcement learning and synergistic control of the act hand. *IEEE/ASME Transactions On Mechatronics*, 18(2):569–577, 2012.

[94] Michael T Rosenstein, James J Collins, and Carlo J De Luca. A practical method for calculating largest lyapunov exponents from small data sets. *Physica D: Nonlinear Phenomena*, 65(1-2):117–134, 1993.

[95] Robert Rosenthal and Ralph L Rosnow. *Artifacts in behavioral research: Robert Rosenthal and Ralph L. Rosnow's classic books*. Oxford University Press, 2009.

[96] Morgan Sangeux, Elyse Passmore, H Kerr Graham, and Oren Tirosh. The gait standard deviation, a single measure of kinematic variability. *Gait & Posture*, 46:194–200, 2016.

[97] Geise Santos, Marcelo Wanderley, Tiago Tavares, and Anderson Rocha. A multi-sensor human gait dataset captured through an optical system and inertial measurement units. *Scientific Data*, 9(1):1–10, 2022.

[98] Shinichi Sato, Masaki Sano, and Yasuji Sawada. Practical methods of measuring the generalized dimension and the largest lyapunov exponent in high dimensional chaotic systems. *Progress of theoretical physics*, 77(1):1–5, 1987.

[99] Martin Schepers, Matteo Giuberti, Giovanni Bellusci, et al. Xsens mvn: Consistent tracking of human motion using inertial sensing. *Xsens Technol*, 1(8), 2018.

[100] Céline Schreiber and Florent Moissenet. A multimodal dataset of human gait at different walking speeds established on injury-free adult participants. *Scientific data*, 6(1):1–7, 2019.

[101] Syed Tafseer Haider Shah and Xiang Xuezhi. Traditional and modern strategies for optical flow: an investigation. *SN Applied Sciences*, 3(3):1–14, 2021.

[102] Abhishek Sharma, Vijeth Rai, Melissa Calvert, Zhongyi Dai, Zhenghao Guo, David Boe, and Eric Rombokas. A non-laboratory gait dataset of full body kinematics and egocentric vision. *Scientific Data*, 10(1):1–11, 2023.

[103] Abhishek Sharma and Eric Rombokas. Complexity of locomotion activities in an outside-of-the-lab wearable motion capture dataset. *Frontiers in Bioengineering and Biotechnology*, 10, 2022.

[104] Abhishek Sharma and Eric Rombokas. Improving imu-based prediction of lower limb kinematics in natural environments using egocentric optical flow. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 30:699–708, 2022.

[105] Astrini Sie, Maxim Karrenbach, Charlie Fisher, Shawn Fisher, Nathaniel Wieck, Callysta Caraballo, Elisabeth Case, David Boe, Brittney Muir, and Eric Rombokas. Descending 13 real world steps: a dataset and analysis of stair descent. *Gait & Posture*, 92:383–393, 2022.

[106] 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.

[107] Karen Simonyan and Andrew Zisserman. Two-stream convolutional networks for action recognition in videos. *arXiv preprint arXiv:1406.2199*, 2014.

[108] Sibo Song, Vijay Chandrasekhar, Bappaditya Mandal, Liyuan Li, Joo-Hwee Lim, Giduthuri Sateesh Babu, Phyo Phyo San, and Ngai-Man Cheung. Multimodal multi-stream deep learning for egocentric activity recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 24–31, 2016.

[109] Katherine M Steele, Adam Rozumalski, and Michael H Schwartz. Muscle synergies and complexity of neuromuscular control during gait in cerebral palsy. *Developmental Medicine & Child Neurology*, 57(12):1176–1182, 2015.

[110] Katherine M Steele, Matthew C Tresch, and Eric J Perreault. Consequences of biomechanically constrained tasks in the design and interpretation of synergy analyses. *Journal of neurophysiology*, 113(7):2102–2113, 2015.

[111] Fernando Daniel Suárez. Čech cohomology and covering dimension for the h maximal ideal space. *Journal of functional analysis*, 123(2):233–263, 1994.

[112] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8934–8943, 2018.

[113] Frank Sup, Amit Bohara, and Michael Goldfarb. Design and control of a powered transfemoral prosthesis. *The International journal of robotics research*, 27(2):263–273, 2008.

[114] Guy Tevet, Sigal Raab, Brian Gordon, Yonatan Shafir, Amit H Bermano, and Daniel Cohen-Or. Human motion diffusion model. *arXiv preprint arXiv:2209.14916*, 2022.

[115] Henry C Thode. *Testing for normality*. CRC press, 2002.

[116] Ene-Margit Tiit. Nonparametric statistical methods. myles hollander and douglas a. wolfe, wiley, chichester, 1999. no. of pages: xiii+779. price: £ 39.95. isbn 0-471-19045-4. *Statistics in Medicine*, 19(10):1386–1388, 2000.

[117] Lena H Ting and J Lucas McKay. Neuromechanics of muscle synergies for posture and movement. *Current Opinion in Neurobiology*, 17(6):622–628, 2007. Motor systems / Neurobiology of behaviour.

[118] 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.

[119] 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.

[120] 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.

[121] Ding Wang, Lin Du, and He Huang. Terrain recognition improves the performance of neural-machine interface for locomotion mode recognition. In *2013 International Conference on Computing, Networking and Communications (ICNC)*, pages 87–91, 2013.

[122] Xuanhan Wang, Lianli Gao, Jingkuan Song, Xiantong Zhen, Nicu Sebe, and Heng Tao Shen. Deep appearance and motion learning for egocentric activity recognition. *Neurocomputing*, 275:438–447, 2018.

[123] William H Warren, Bruce A Kay, Wendy D Zosh, Andrew P Duchon, and Stephanie Sahuc. Optic flow is used to control human walking. *Nature neuroscience*, 4(2):213–216, 2001.

[124] 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.

[125] Samuel S Wilks. Certain generalizations in the analysis of variance. *Biometrika*, pages 471–494, 1932.

[126] 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.

[127] Shuen-De Wu, Chiu-Wen Wu, Shiou-Gwo Lin, Chun-Chieh Wang, and Kung-Yen Lee. Time series analysis using composite multiscale entropy. *Entropy*, 15(3):1069–1084, 2013.

[128] 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.

[129] Fan Zhang, Ming Liu, and He Huang. Effects of locomotion mode recognition errors on volitional control of powered above-knee prostheses. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 23(1):64–72, 2014.

[130] Fan Zhang, Tingfang Yan, and Max Q.-H. Meng. Gait phase recognition based on a wearable depth camera*. In *2018 IEEE International Conference on Information and Automation (ICIA)*, pages 756–760, 2018.

[131] 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.

[132] 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.

[133] 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.

[134] Chaobin Zou, Rui Huang, Hong Cheng, and Jing Qiu. Learning gait models with varying walking speeds. *IEEE Robotics and Automation Letters*, 6(1):183–190, 2020.

Appendix A

RESOURCES

- Dataset collected as a part of this dissertation is available on the following links:

  Part 1: `https://doi.org/10.6084/m9.figshare.c.6076607.v1`,

  Part 2: `https://doi.org/10.25452/figshare.plus.21761465.v1`

- The code for usage of the dataset and some example applications are available on the following github repository:

  `https://github.com/abs711/The-way-of-the-future`

- The code related to environment modeling (chapter 3) is available on the following repository:

  `https://github.com/abs711/Visual-Control`

- The code for personalization using style transfer (chapter 5) will be made available on the following repository:

  `https://github.com/abs711/Optimizing-representations-for-gait-generation`

- For future work updates please check my website and github:

  Website: `https://abs711.github.io/`

  Github: `https://github.com/abs711`