Mode-free Control of Prosthetic Lower Limbs

1st Vijeth Rai
Dept of Electrical Engineering
University of Washington
Seattle, WA, USA.
raiv@uw.edu

2nd Abhishek Sharma
Dept of Mechanical Engineering
University of Washington
Seattle, WA, USA
as711@uw.edu

3rd Eric Rombokas
RR&D Center for Limb Loss and MoBility (CLiMB)
VA Puget Sound Health Care System
Seattle, WA, USA
rombokas@uw.edu

***Abstract*—Intuitive control of powered prosthetic lower limbs is still an open-ended research goal. Current controllers employ discrete locomotion modes for well-defined scenarios such as stair ascent, stair descent, or ramps. General human locomotion, however, is a continuous motion, fluidly adapting to the environment, and not always categorizable into modes. A prominent feature of normal movement is that it exhibits strong inter-joint coordination, and the movement of a single joint, such as the ankle, can be largely predicted based on the movement of the rest of the body. We show that using body motion from the intact limbs and trunk, a reference trajectory can be generated for a prosthetic joint for every instant in time.**

A wearable motion capture suit was worn by 11 healthy subjects to record full body kinematics during flat ground walking with and without random stops and stair ascent and descent. Three machine learning techniques were employed to predict right ankle kinematic trajectory using other joint kinematics as inputs. We found that a Recurrent Neural Network (RNN) was the best performing model, robust to subject-specific variations such as walking speed and step length. Performance of the network evaluated on a test subject (not included in training) showed that ankle angle can be estimated with a root mean square error of fewer than 7 degrees. The change in performance when using a smaller array of sensors, gathering partial body kinematics instead of full-body motion, is also evaluated. This approach demonstrates the potential for the application of data-driven models to prosthetic control without explicit featurization of terrains or gait phases.

***Index Terms*—prosthetic limbs, deep learning, gait analysis, motion capture**

I. Introduction

Prosthetic lower limbs are becoming more complex, and some include active power and variation of multi-dimensional dynamic properties [1]. These increasingly robotic devices have the potential to dramatically improve mobility. The mechanical and design challenges are increasingly being addressed, but controlling the devices remains a significant challenge [2].

Most powered prosthetic lower limbs have modes of operation corresponding to terrain types, such as flat-ground walking or stair ascent. Control is achieved by choosing the appropriate mode-specific sub-controller, which generates a predefined trajectory command (position, torque). The current state-of-the-art prosthetic controllers apply mechanical sensors, EMG sensors or a combination of the two to gauge user intent to select the right mode for the current terrain [3], [4]. This strategy has successfully enabled users of powered prosthetic limbs to navigate typical structured scenarios with more natural gait than ever before possible using passive prostheses. At this stage, navigation of unstructured situations, such as obstacle avoidance or rough terrain, are ready to be addressed.

Human locomotion is a continuous control process, difficult to categorize into discrete modes. Humans optimize and select complex unique movements based on past experiences of navigating similar conditions. To emulate such natural control for powered prostheses would necessitate capturing and replaying these complex movements given a similar scenario. However, the normative human movement has been hard to capture for real-world behavior on unstructured terrain. Furthermore, selecting the appropriate example among the multitude of possible movements is analogous to a very high dimensional policy search - the intractability of which has contributed to using locomotion modes to simplify the control problem.

Towards natural mode-free control of lower-limb prosthesis, a promising solution is motivated by the observation that physiological gait exhibits strong inter-joint coordination. The trajectory of movements of the intact limbs could provide the means to estimate the movement of the missing limb with a high likeness to the movements observed in real scenarios, without resorting to modes.

An earlier approach called Echo control employed replaying or "echoing" the movement of the sound leg on the prosthetic limb [5]. This delayed playback method failed when asymmetric movements are desired and also required the sound leg to lead all movements.

Vallery et al [6] demonstrated an approach that addressed these shortcomings. Instead of simply replaying the movements of the contralateral limb, they obtain a linear mapping to estimate the motion of the impaired limb. Their Complementary Limb Motion Estimation (CLME) method used statistical regression to show that instantaneous mode-free control is possible. However, it did not factor in the time-history of gait, which may be necessary to emulate more complex movements.

Data-driven approach

Recent successes in machine learning have leveraged the availability of large datasets rather than employing carefully human-engineered rules. This learn by examples method has surpassed all previous learn by rules efforts in a variety of applications, such as speech recognition, and stock-market prediction [7]. To apply a data-driven approach for locomotion control of prosthetic limbs would require whole body motion sensing to capture a copious amount of gait data of unstructured scenarios, along with computing power to manage such a high dimensional and high volume data-set.

Recently, it has become possible to use portable motion capture suits that provide performance rivaling traditional laboratory-grade motion capture. While optical motion capture in a camera instrumented gait analysis laboratory remains the gold standard, body-worn devices that track the users motion continue to improve, becoming cheaper, smaller, more efficient, and less intrusive. In our study, we use a commercially available motion tracking suit from Xsens to capture full body kinematics (joint angles). We anticipate that wearable accessories with sensors will be ubiquitously available in the future enabling easy acquisition of such full body kinematics data in real-time.

Here we present a proof of concept for predicting ankle movement given kinematic measurements of the rest of the body. We explore how well walking dynamics can be translated across individuals with self-selected walking speed for 3 different machine learning approaches:

- Linear Regression with no time-history.
- Fully-Connected Deep Neural Network with time-history of gait data.
- Recurrent Neural Network with time-history of gait data.

We also explore the contribution of different sensors. We collected full sensor data, but we present the performance for ankle prediction using only Sagittal plane data and using only lower limb kinematics.

The ultimate goal of this line of research is to provide better control for unstructured scenarios. However, such a controller should also be applicable to more commonly encountered situations. We address the more commonly encountered scenarios of flat ground and stairs in this work, leaving more unstructured complex scenarios as for future work, once a suitable framework has been established.

In Section II we describe the experimental setup, data collection, processing, and the machine learning methods employed. We also discuss the analysis of predictions and optimization performed to choose the best model and framework. We report the results of these analyses in Section III and discuss their implications in Section IV.

II. Methods

A. Participants

Ambulation data was collected for 11 healthy participants (7 males, median age of 25) with no amputation or other mobility impairments. The experiment was completed in a single session which lasted less than 2 hours. Recruitment and human subject protocols were performed in accordance with local University of Washington Institutional Review Board approval and each subject provided informed consent. De-identified data can be made available, via a data use agreement, upon request to the authors.

B. Experiments

Each subject performed flat-ground walking, flat-ground walking with random stops, and stair ascent and descent, for 10 to 15 minutes using a self-selected pace. Subjects were instructed to walk naturally and the data was collected in public spaces during active business hours, with the intent that normal gait dynamics and corrections would appear in the example data. The order of activities, starting and ending points were randomized. This allowed a range of variability across subjects such as course length, duration, number of upstairs and downstairs, etc. Stair ascent and descent included short segments of flat walking on the stair landings. In total, 300 mins of data were collected from all subjects.

C. Instrumentation

We collected locomotion data using the Xsens Awinda suit [8], which consists of 17 body-worn sensors placed at key locations. Each sensor has a 3-axis gyroscope, accelerometer, magnetometer, and barometer. Xsens Analyse software was used to integrate these individual sensors and render a full-body avatar for confirming successful placement and calibration. After a system specified calibration method was performed, the software provided position and joint kinematics in a 3D environment. Although other data such as limb-segment position, orientation, acceleration are available, we used only joint angles for this study. All angles are in 1x3 Euler representation of the joint angle vector (x, y, z) in degrees, calculated using the Euler sequence ZXY using the International Society of Biomechanics standard joint angle coordinate system [9].

Data from a total of 22 joints in 3 anatomical planes (Sagittal, Frontal, Transverse) were captured for each trial, resulting in 66 total possible features, 60 times per second.

D. Data Processing

1) Initial: Sensor data was continuously visually inspected to see any aberrant errors. During data collection, some sensors might get displaced from their original calibrated location. If detected during the experiment, sensor placement was corrected followed by re-calibration and re-initialization of the suit.

2) Normalization and Reshaping: Each of the joint angles exhibits a different Range of Motion (ROM). In order to prevent high-ROM joints from dominating predictions, it is common practice to normalize all features (generally 0 to 1). We normalized all joint angles for every trial and saved the scaling factor for recovering the original signal. Data from all trials and subjects were stitched together into one combined dataset of shape $[num\_of\_samples, num\_of\_features]$.

Inputs were further segmented into a rolling window prior to training the neural net described later in Section II-F0a. The number of samples varied for different activities but overall we collected approximately 1080K samples at a rate of 60Hz. The number of features also varied with different analysis scenarios (see Section II-G).

As our target feature was right ankle trajectory in the Sagittal plane, all 3 features pertaining to right ankle (Sagittal, Frontal and Transverse planes) were omitted from the training set. One of the subjects was omitted from the training set. A random trial from this subject was used as the validation set, and another was used as the test set. Predicted ankle angle was denormalized and root mean squared error was reported.

E. Machine Learning Models

Complementary Limb Motion Estimation (CLME) [6] applied linear regression to find a static mapping from contralateral hip and knee to an actuated knee joint. This showed great promise for using limb coordination for predicting a target joint angle, but used a relatively simple computational method and did not take the time history of motion into account. In order to compare the present method to CLME, we analyzed performance on our dataset using multivariate linear regression, as well as two neural network architectures that accommodate the time-varying history.

1) Linear Regression: A linear regression model from the input data was created using the MATLAB's *fitlm* function. No time-history was considered for the linear model; each instant of time can be used to generate a prediction, computed independently of one another. This least squares approach is considered the Best Linear Unbiased Estimator [10].

2) Fully-Connected Deep Neural Network (DNN): The neural network architectures were implemented on Google's Tensorflow framework [11]. In order to compare performance against a simpler neural network, the first model was a Fully-Connected Deep Neural Network (DNN). This is simply a modern implementation of the well-known neural networks that have been used for some time, with updates that have greatly improved their training and performance. Though a fully-connected network does not explicitly compute a time series, it is possible to train by presenting a fixed window of time history as the input to the network. Although the network is seeing all the inputs in a time sequence at once, it doesn't necessarily see the temporal and causal relationship between inputs in the right order.

3) Recurrent Long Short-Term Memory Neural Network (LSTM): Recurrent networks, by contrast, are trained on a sequence, including their previous outputs to earlier inputs in the sequence. We use a particularly powerful variant, the Long Short-Term Memory (LSTM) [12]. The network consisted of several LSTM layers, composed of cells that learn to predict the current ankle angle given a sequence of whole-body joint angles (Figure 1). LSTMs are of particular interest to our application due to their ability to learn temporal dependencies better by efficiently "remembering" valid predictors from past inputs and "forgetting" unnecessary artifacts in the data.

F. Loss and Parameter Optimization for Neural Networks

The neural networks are capable of taking a sequence of input data and hence required a few additional data processing steps described below. They also have several hyperparameters that need to be optimized for different application domains.

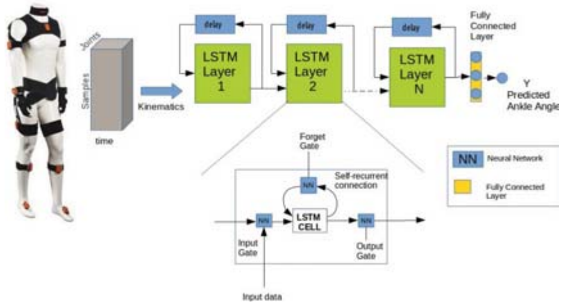

Fig. 1. Recurrent Neural Network Architecture used.

a) Time-Window: LSTMs differ in their implementation in that they backpropagate the errors a specific number of time steps back. This parameter, known as the sequence length, affects the time scale that the RNN cell state reasons about. Choosing too long of a sequence length would need more parameters to be retained, increasing computational load and decreasing the network efficiency of learning the appropriate predictors in the data. Too short of a sequence length increases the difficulty of learning time dependencies in the data.

We implement a rolling window of time series data, such that the continuous stream generated in Section II-D is segmented into samples of desired sequence length. Hyperparameter optimization and evaluations are performed on this windowed data. We then vary the sequence length and repeat the same. The average performance of each is recorded to evaluate the best sequence length or duration of the time history.

b) Loss and Optimizer: The loss function to be optimized for the neural network was the mean squared error (MSE) between the predicted $\hat{y}$ and actual y ankle angle. An L2 regularization loss was also included in the loss function. This loss penalizes large weights of the network, allowing all features to be considered and avoiding overfitting. The overall impact of the regularization term in the total loss function is tuned using the regularization rate parameter λ. The optimal λ was obtained during Hyperparameter optimization using the Adam [13] optimizer.

c) Hyperparameter Optimization: A combination of random and grid search was applied to optimize hyperparameters. Training inputs consisting of time-segmented samples (Section II-F0a) were divided into mini-batches. Each batch was shuffled and random Gaussian noise was added to each sample to reduce over-fitting.

Parameters optimized for LSTM network include batch size, number of epochs, number of layers (L), number of units in each layer (HU), the standard deviation of the injected noise, the regularization parameter for L2 loss (λ) and, learning rate. Parameters optimized for DNN include the number of fully connected layers and the number of units in each layer.

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. 10

Hyperparameter	Range/Values	Optimal
Learning Rate	$[10^{-5} : 10^{-2}]$	$5 * 10^{-3}$
Batch Size	[32,64,512]	512
Number of Epochs	[20,50,100,200]	50
Input Sequence Length	[1,2,5,10,15,20,40]	10
Number of LSTM Layers	[1,2,4]	2
Number of Hidden Units	[4,8,16,32]	32
Regularization Rate	[0.015,0.025]	0.025
Standard Deviation of Random Noise	[0.01,0.02]	0.01

TABLE I
HYPERPARAMETER VALUES TESTED FOR OPTIMAL PERFORMANCE ON STAIRS DATA-SET

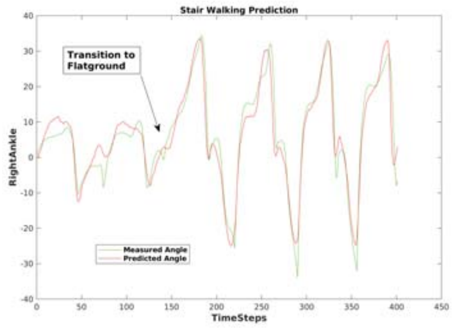

Fig. 2. LSTM model predictions for stair ascent and descent activity with an average RMSE of 4.4 degrees. The LSTM was the best performing machine learning model.

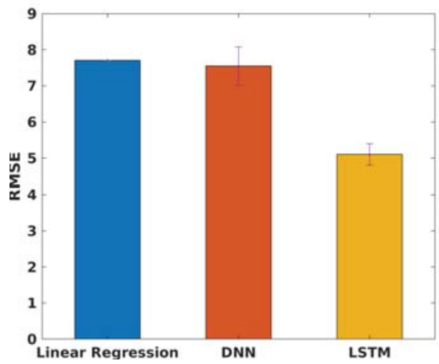

Fig. 3. Error for each class of machine learning algorithm. Both neural network models showed improved performance, with the Recurrent Neural Network (LSTM) performing best by a large margin.

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. For example, even though training the models on 200 epochs occasionally performed better than 50 epochs, the improvement was negligible enough to select 50 as the optimal number of epochs. The range of parameter values tested is shown in Table I. The optimal hyperparameter set was used to compare and evaluate performance.

G. Analysis

To address the practical aspect of using our method, we investigate performance with respect to varying scenarios. We performed four types of comparisons.

To apply this technique reliably with minimum instrumentation, we explore prediction performance using different sensor groups. We explore performance for different activities with different inherent dynamics and complexities. As this control strategy relies heavily on data, we also investigated performance dependence on varying amount of data.

1) *Machine Learning Model:* We compared machine learning models:
   - Multivariate Linear Regression
   - Fully-Connected Deep Neural Network
   - Recurrent Neural Network - LSTM
2) *Sensor Groups:* We quantify performance for the following sensor groups:
   - Full body (20 joints) in all 3 anatomical planes
   - Full body, only Sagittal plane
   - Lower Limb (6 joints) in all 3 anatomical planes
   - Lower Limb, only Sagittal plane.
3) *Activities:* We compared performance for model trained on data from:
   - Flatground walking only
   - Flatground walking with random stops only
   - Stairs with up and down stairs only
   - Combined dataset of all activities.
4) *Dependence on data*. We assessed performance for different amount of training data by varying the following two parameters:
   - Number of subjects included in training data.
   - Percentage of data included from *every* subject in training data.

III. RESULTS

The performance of the right ankle predictions, trained as described above, was assessed in terms of Root Mean Squared Error (RMSE) between the predicted and measured angle in the test set. An example of a typical prediction for stair climb activity using the LSTM model, with an RMSE of 4.4 degrees is shown in Figure 2.

A. Machine Learning

Performance for each of the types of machine learning models is shown in Figure 3. Both of the neural network models showed improved performance over linear regression. The LSTM network showed a more dramatic improvement.

B. Sensor Groups

The performance was best when full body kinematics in all three anatomical planes were used for training. However, the loss in performance was small when using only the lower limb sensors (Figure 4). Within each group, performance dropped when only the Sagittal plane was considered. Stairs data are used as a representative example in Figure 4, but the trend holds across all activities.

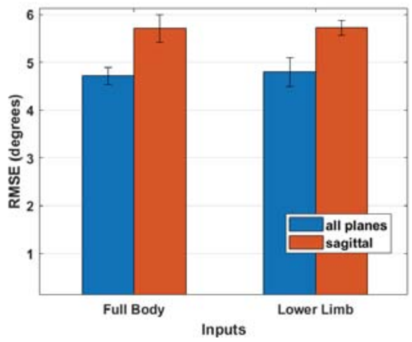

Fig. 4. Performance for subsets of sensors. Considering data from all 3 anatomical planes (Frontal, Transverse and Sagittal) improved performance. Using only the lower limb sensors for training did not result in significant drop in performance.

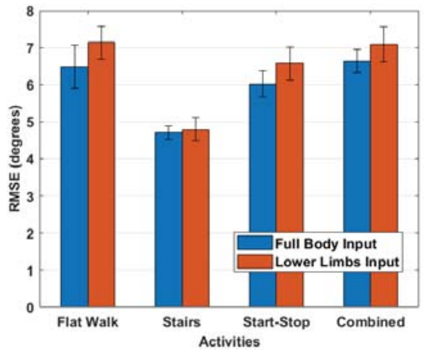

Fig. 5. Error for models trained on specific activities or all combined. Performance was best for the Stair Ascent and Descent controller. The network model trained on all activities resulted in a qualitatively similar performance as the worst performing single activity model.

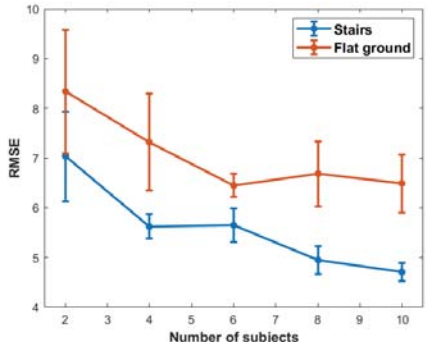

Fig. 6. Error for models trained on different number of subjects. Performance improved with more subjects included in training data

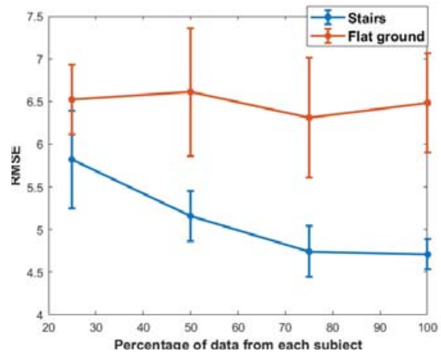

Fig. 7. Error for models trained using partial data from every subject. Performance was relatively stable as long as the network was exposed to some part of every subject's data. Models trained on 50 percent and 100 percent of the data were approximately similar in performance

C. Activities

Activities dramatically impacted performance of prediction. Surprisingly, stair ascent and descent was the activity with the best performance, even though flat walking might be assumed to be a kinematically simpler activity (Figure 5). It is possible that this is due to flat ground walking allowing more variability of step length and frequency than stairs. Interestingly, the LSTM network model that was trained on all of the activities together had qualitatively similar performance to the activity-specific ones, resembling the worst single-activity controller but not worse. This is a promising result for the overall goal of making a general-purpose mode free control system.

D. Dependence on Data

The primary analysis of performance dependence on the amount of data varied the number of subjects included in the training data. Secondary analysis performed included all the subjects but varied the percentage of data used from individual subject. Prediction error generally decreased with data from more subjects included in training the models (Figure 6). Interestingly, the error remained approximately the same even when half the data from all subjects was not included in the training (Figure 7). This suggests that data collection could be designed to favor shorter sessions from more number of subjects over longer sessions from fewer subjects.

IV. Discussion

The simplifying assumption of gait modes is not motivated by the appropriateness of modes to human motion, but by the fact that an all-purpose controller has not been possible. Similarly, powered prosthetic devices do not incorporate full body motion not because that information is useless, but because it has not been previously practical to measure. This study shows that these limitations are disappearing and provides a distinct alternative to current state-of-the-art assistive device controllers. Using only kinematic data from non-invasive sensors, we showed that ankle joint trajectory can be continuously estimated even for acyclic gait events, for every instant in time. These predictions would be the core of a mode-free intuitive user intent estimation system that coordinates the movements of the assistive device with those of the body. This mode-free strategy provides the capability to generate movement patterns that might be difficult for us to explicitly model, but that we can measure many examples of.

Key findings in this study

The performance was improved when more subjects were included and was best when all the subjects (except test subject) were included in the training data. This peak performance was maintained even when half the data from *every* subject was discarded (Figure 6 and 7). These results have two-fold implications for more efficient data collection protocols in the future. Firstly, as expected, a data-driven approach relies on and benefits by including data from more subjects. Secondly, less data from more number of subjects is better than more data from fewer subjects.

An interesting result is that although usage of full body sensors gave the best results, the drop in performance when considering only lower-limb sensors was insignificant (Figure 4). However, using only Sagittal plane data showed a more noticeable degradation of performance. This suggests that, in the interest of minimum instrumentation and cost, such a control strategy could be deployed with just lower-limb sensors without compromising overall performance.

The recurrent long short-term memory network was the best performing model. The LSTM learns an internal model that selectively retains useful previous inputs while ”forgetting” the unnecessary artifacts. Although LSTMs are computationally heavier than some other methods and therefore take significantly more time to train, the improvement of performance justifies their use. Once trained, these networks are fast to deploy. Several new embedded processors [14] are entering the market specifically geared towards fast inference from previously trained models.

Our optimization process showed that adding time-history and hidden units improved performance but with diminishing returns. RMSE dropped significantly for sequences longer than 2 time samples (32ms) but plateaued after 10 samples (160ms). The additional hidden units give better expressibility, however, only to a certain extent. More time-history and hidden units also increase the computational load.

A clear benefit of a data-driven controller is the possibility it provides to adapt and upgrade to accommodate more activities, just like new experiences allow us to learn new locomotor skills. Figure 5 demonstrates that a controller trained on combined activities performs as well as the worst single-activity controller. As data that include new activities are collected, new models can be trained offline and downloaded onto the device, allowing the user to perform the new tasks.

Comparison with other studies

Prior research has explored the application of kinematic data for control of prosthesis. The research groups in [15], [16] used a combination of EMG and inertial data along with machine learning techniques to classify human gait. Both these groups use gait events of heel-strike and toe off to segment gait into temporal windows and the trained classifiers output a categorical class label for every window corresponding to the particular mode. The method demonstrated here requires neither modes nor explicit segmentation of gait events.

Using only kinematics data from noninvasive sensors also has an additional benefit of not relying on myoelectric signals which have been known to be highly temperamental [17]. This could reduce the load on prosthetists to find appropriate sites for placement of EMG electrodes.

Given that our goal is to eliminate the categorical classification of gait, the performance metric we use is RMSE which is appropriate for regression. This makes it difficult to compare results reported here with prior studies, which use locomotion mode classification accuracy as the metric. Experiments in preparation will be geared towards assessing the overall performance of this control strategy as well as to subjectively compare it to other mode-based control strategies. to provide

Future work

In this paper, we explore a moderate activity set. Evaluation of obstacle course data and unstructured real-world scenarios will be tackled in future work.

Safety and recovery from perturbations are crucial for any lower limb prosthetic controller. An error in upper-limb control strategy can be inconvenient but can be fatal in the case of lower-limb. Humans recover from perturbations using very unique balancing maneuvers. The current mode-based controllers fail to capture these whole body nuances and generally have to resort to very awkward resets to recover. One of the intended future datasets would include reactions to unexpected perturbations and fall-recovery maneuvers.

We show that deep networks can represent and adapt to subjects not directly measured during training. So even though the prosthetic user's own gait may not have been recorded prior to amputation, this approach could still be employed. However, a major assumption with this approach is that normative trajectories collected from unimpaired subjects can be replayed for prosthetic users. Although there is evidence that gait patterns are similar for subjects with same hip height [18] it remains to be seen if this holds true when other factors, such as the weight and the contribution of energy, of the missing limb are different from the intact one.

V. Conclusion

This manuscript presents a baseline for prediction performance and establishes a framework for the control strategy. This demonstrates the potential of our novel technique but many questions remain before it can be deployed. Future experiments will transition these joint trajectory prediction network models to real time prosthetic controllers. Implementation details, such as detecting when the user is adjusting a sensor and not interpreting it as a movement remain before the system should be extensively user tested. We are now refining the system to improve prediction, and have begun preliminary tests using a live, powered prosthesis.

References

[1] M. Goldfarb, H. A. Varol, F. C. Sup IV, J. Mitchell, and T. J. Withrow, “Powered leg prosthesis and control methodologies for obtaining near normal gait,” 2015.

[2] M. R. Tucker, J. Olivier, A. Pagel, H. Bleuler, M. Bouri, O. Lambercy, J. del R Millán, R. Riener, H. Vallery, and R. Gassert, "Control strategies for active lower extremity prosthetics and orthotics: a review," *Journal of neuroengineering and rehabilitation*, vol. 12, no. 1, p. 1, 2015.

[3] L. J. Hargrove, A. J. Young, A. M. Simon, N. P. Fey, R. D. Lipschutz, S. B. Finucane, E. G. Halsne, K. A. Ingraham, and T. A. Kuiken, "Intuitive control of a powered prosthetic leg during ambulation: a randomized clinical trial," *JAMA*, vol. 313, no. 22, pp. 2244–2252, 2015.

[4] H. A. Varol, F. Sup, and M. Goldfarb, "Multiclass real-time intent recognition of a powered lower limb prosthesis," *Biomedical Engineering, IEEE Transactions on*, vol. 57, no. 3, pp. 542–551, 2010.

[5] D. L. Grimes, "An active multi-mode above knee prosthesis controller," Ph.D. dissertation, Massachusetts Institute of Technology, 1979.

[6] H. Vallery, R. Burgkart, C. Hartmann, J. Mitternacht, R. Riener, and M. Buss, "Complementary limb motion estimation for the control of active knee prostheses," *Biomedizinische Technik/Biomedical Engineering*, vol. 56, no. 1, pp. 45–51, 2011.

[7] A. Halevy, P. Norvig, and F. Pereira, "The unreasonable effectiveness of data," *IEEE Intelligent Systems*, vol. 24, no. 2, pp. 8–12, 2009.

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

[9] G. Wu, S. Siegler, P. Allard, C. Kirtley, A. Leardini, D. Rosenbaum, M. Whittle, D. D DLima, L. Cristofolini, H. Witte *et al.*, "Isb recommendation on definitions of joint coordinate system of various joints for the reporting of human joint motionpart i: ankle, hip, and spine," *Journal of biomechanics*, vol. 35, no. 4, pp. 543–548, 2002.

[10] S. Puntanen and G. P. Styan, "The equality of the ordinary least squares estimator and the best linear unbiased estimator," *The American Statistician*, vol. 43, no. 3, pp. 153–161, 1989.

[11] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner, I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, "TensorFlow: Large-scale machine learning on heterogeneous systems," 2015, software available from tensorflow.org. [Online]. Available: https://www.tensorflow.org/

[12] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[13] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[14] "What makes tpus fine-tuned for deep learning? — google cloud blog," https://cloud.google.com/blog/products/ai-machine-learning/what-makes-tpus-fine-tuned-for-deep-learning, (Accessed on 10/24/2018).

[15] J. D. Miller, M. Seyedali, and M. E. Hahn, "Walking mode classification from myoelectric and inertial fusion," in *ASME 2012 Summer Bioengineering Conference*. American Society of Mechanical Engineers, 2012, pp. 797–798.

[16] B. H. Hu, E. J. Rouse, and L. J. Hargrove, "Using bilateral lower limb kinematic and myoelectric signals to predict locomotor activities: A pilot study," in *Neural Engineering (NER), 2017 8th International IEEE/EMBS Conference on*. IEEE, 2017, pp. 98–101.

[17] D. Farina, R. Merletti, B. Indino, M. Nazzaro, and M. Pozzo, "Surface EMG crosstalk between knee extensor muscles: experimental and model results," *Muscle & nerve*, vol. 26, no. 5, pp. 681–695, 2002.

[18] M. P. Murray, "Gait as a total pattern of movement: Including a bibliography on gait," *American Journal of Physical Medicine & Rehabilitation*, vol. 46, no. 1, pp. 290–333, 1967.