Subject-Independent, Biological Hip Moment Estimation during Multimodal Overground Ambulation using Deep Learning

Dean D. Molinaro*, *Student Member, IEEE*, Inseung Kang, *Student Member, IEEE*, Jonathan Camargo, *Student Member, IEEE*, Matthew C. Gombolay, *Member, IEEE,* and Aaron J. Young

***Abstract*—Estimating biological joint moments using wearable sensors could enable out-of-lab biomechanical analyses and exoskeletons that assist throughout daily life. To realize these possibilities, this study introduced a subject-independent hip moment estimator using a temporal convolutional network (TCN) and validated its performance and generalizability during multimodal ambulation. Electrogoniometer and simulated IMU data from sixteen participants walking on level ground, ramps and stairs were used to evaluate our approach when benchmarked against a fully-connected neural network, a long short-term memory network, and a baseline method (i.e., using subject-average moment curves based on ambulation mode and gait phase). Additionally, the generalizability of our approach was evaluated by testing on ground slopes, stair heights, and gait transitions withheld during model training. The TCN outperformed the benchmark approaches on the hold-out data ($p<0.05$), with an average RMSE of 0.131±0.018 Nm/kg and R^2 of 0.880±0.030 during steady-state ambulation. When tested on the 20 leave-one-out slope and stair height conditions, the TCN significantly increased RMSE only on the steepest (+18°) incline ($p<0.05$). Finally, the TCN RMSE and R^2 was 0.152±0.027 Nm/kg and 0.786±0.055, respectively, during mode transitions. Thus, our approach accurately estimated hip moment and generalized to unseen gait contexts using data from three wearable sensors.**

***Index Terms*— Biomechanics estimation, machine learning, multimodal ambulation, neural networks, wearable sensors**

I. Introduction

LOWER limb joint moment estimation using wearable sensors could enable out-of-lab biomechanical analyses, provide real-time joint dynamics for long-term health monitoring, and could be used to modulate exoskeleton assistance during a variety of daily activities. Conventionally, biological joint moments are computed using inverse dynamics of subject-specific anatomical models [1], which rely on motion capture and ground reaction force measurements from stationary, in-lab systems (i.e. motion capture systems and force plates). This method has limited biomechanical analyses to space- and time-constrained studies and mitigates the adoptability of biomechanically informed clinical interventions. Additionally, exoskeleton controllers typically rely on hand-engineered gait variables (e.g., ambulation mode and gait phase) to modulate assistance [2]–[6]. These approaches have successfully augmented human ambulation [7]–[10] but have been primarily limited to steady-state gait, such as constant speed treadmill walking. Currently, it is unknown how these robotic systems would affect human gait during real-world ambulation, especially during gait that is not well-defined by these hand-engineered variables.

To reduce these limitations inherent to previous biomechanical analyses and exoskeleton controllers, researchers have developed biological joint moment estimators using anatomical, neuromusculoskeletal, and machine learning models informed by wearable sensors. Of these methods, machine learning approaches have the greatest potential to remove the need for subject-specific calibration and to reduce sensor suite complexity; however, model generalizability to a variety of ambulation conditions remains a concern. For instance, Stetter *et al*. found that a fully-connected neural network (FCNN) did not maintain accurate knee moment estimates when trained and tested across a wide variety of tasks [11]. Further, previous research has not evaluated model generalization to ambulatory conditions withheld from training, despite its relevance for real-world implementation.

To address these limitations, we developed a novel joint moment estimator using a temporal convolutional network (TCN) [12] and evaluated its ability to estimate the biological hip flexion/extension moment of novel subjects across a wide variety of ambulatory tasks. These tasks included level ground, ramp ascent/descent, and stair ascent/descent ambulation with varying walking speeds, slopes, and stair heights. Since machine learning models can estimate biological joint moments using reduced sensor information, we limited the input of the TCN model to data from trunk and thigh inertial measurement units (IMUs) and a sagittal hip goniometer. We compared our

This work was supported by the National Science Foundation Graduate Research Fellowship under Grant No. DGE-1650044 and the NSF NRI Award No. 1830215. This work was also supported by the Fulbright Foreign Student Fellowship.

*D. D. Molinaro, J. Camargo, M. C. Gombolay, and A. J. Young are with the Institute for Robotics and Intelligent Machines (IRIM), Georgia Institute of Technology, Atlanta, GA (correspondence email: dmolinaro3@gatech.edu). D. D. Molinaro, I. Kang, J. Camargo, and A. J. Young are also with the George W. Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA. M. C. Gombolay is with the School of Interactive Computing, Georgia Institute of Technology, Atlanta, GA.

approach to three alternative hip moment estimation methods: a FCNN, a long short-term memory network (LSTM), and a non-machine learning baseline method. The baseline method estimated instantaneous hip moment from subject-averaged, mode-specific hip moment curves computed from the ground-truth labels, which is an approach derived from biologically-inspired torque controllers for wearable robots [8], [9]. The hyperparameters of each neural network were optimized to the task set to ensure a fair comparison among the estimators.

We hypothesized that the TCN would improve the estimation root-mean-square error (RMSE) and coefficient of determination (R^2) compared to the alternative neural networks as the TCN used dilated, causal convolutional layers to encode temporal information and learn features from the input data, which has led to competitive performance of TCN models in many previous sequence modeling tasks [12]. Additionally, we hypothesized that our approach would improve estimation RMSE and R^2 when compared to the baseline method, since the inputs to the TCN contain stride-specific data. Finally, we evaluated the ability of the TCN to estimate hip moments during ambulation mode transitions, ramp slopes, and stair heights withheld from the training set to quantify the ability of the model to generalize to unseen conditions common in daily ambulation. Therefore, our study introduces a novel biological hip moment estimator and provides the first comprehensive analysis of such a system to estimate biological hip moments during level ground, ramp, and stair ambulation using a simple kinematic sensor suite. Additionally, our study is the first to quantify the generalizability of a joint moment estimation model to unseen ambulatory contexts, which is an important consideration for the implementation of these systems. In general, our approach accurately estimated biological hip moments and generalized well to unseen gait contexts, indicating our approach is applicable for real-world contexts.

II. Background

Multiple approaches have been developed to estimate biological joint moments using wearable sensors. One such approach is to directly compute biological joint moments using inverse dynamics based on wearable sensor data instead of conventional motion capture and force plate data [13]–[18]. Biological joint kinematics can be measured using electrogoniometers or rotary encoders located at each joint [19] or computed using accelerometer and gyroscope data [15]–[17], [19]. Additionally, the ground reaction forces and moments (GRFs) can be estimated using instrumented footwear (e.g., pressure insoles) [13], [15], [20]–[22], model-based approaches [14], [17], [18], and/or data-driven estimators [21]–[24]. One benefit of the wearable sensor-based inverse dynamics approach is that it is highly generalizable across activities, including walking [13]–[15], [17], running [18] jumping [18], and skiing [16]. Unfortunately, one major limitation of the inverse dynamics approach is that it requires complete kinematic and kinetic data of the anatomical model. This requirement can lead to cumbersome and complicated measurement systems, especially when estimating biological joint moments close to the center of mass (e.g., at the hip joint), limiting the practicality of this method.

Another model-based approach to estimate biological joint moments is to use a forward dynamics neuromusculoskeletal model informed by joint kinematics and electromyography (EMG) [25]–[27]. This approach removes the need for GRF measurements by modeling the muscle force dynamics about the biological joint of interest, using an anatomical model and EMG-driven muscle dynamics [26]. Unfortunately, this approach still requires multi-joint kinematic measurements to account for biarticular muscle dynamics and requires ground-truth biological moments to calibrate the neuromusculoskeletal model given a new subject or EMG electrode placement [25]. Additionally, EMG-informed forward dynamics models require accurate placement of EMG electrodes to model muscle activation dynamics, which requires previous training and detailed setup. These limitations currently reduce the viability of this approach for useful biological joint moment estimation.

Recently, data-driven methods have been implemented as an alternative approach for estimating biological joint moments using reduced sensor information [11], [22], [28]–[32]. Accurate results are encouraging, since these machine learning models could lead to large-scale, out-of-lab biomechanics studies, health monitoring, and unified exoskeleton control, using low-cost, simple sensor suites. For instance, in our previous work, we implemented a FCNN and a XGBoost model that could accurately estimate hip flexion/extension moments during level and sloped walking using only kinematic measurements onboard a hip exoskeleton [29]. Further, Mundt *et al.* recently implemented a FCNN that could accurately estimate lower limb joint moments of a novel subject using inertial data during the stance phase of level walking [30], potentially removing the need for subject-specific calibration. Dorschky *et al.* expanded these findings by introducing a convolutional neural network to estimate sagittal lower limb joint moments of novel subjects during walking and running using inertial input data [32]. Unfortunately, machine learning-based joint moment estimation models may lose relevance when evaluated on additional tasks. For instance, Stetter *et al.* found that training a FCNN to estimate knee moments during a multitude of tasks, including walking, running, turning, and cutting maneuvers, based on IMU data resulted in decreased model performance compared to previous neural networks that were evaluated on a limited task set [11]. This concern prompted our study, which proposed a novel joint moment estimator and evaluated its generalizability to novel subjects and several conditions common in community ambulation.

III. Biomechanical Dataset

A. Experimental Protocol & Measurements

Sixteen able-body participants (10 males, 6 females, height of 1.70±0.07 m, body mass of 68.4±12.2 kg, and age of 22±4 years) provided written informed consent according to the protocol approved by the Georgia Institute of Technology Institutional Review Board for this study. Each participant completed ten circuits of level ground (LG) walking at a slow, self-selected, and fast walking speed (30 total LG circuits per

participant). Each participant also completed 10 trials of ramp ascent (RA), ramp descent (RD), stair ascent (SA), and stair descent (SD) ambulation overground for each ground slope (5.2°, 7.8°, 9.2°, 11°, 12.4°, and 18°) and ADA compliant stair height (10.2 cm, 12.7 cm, 15.2 cm, and 17.8 cm) (Fig. 1). Transitions between each ascent/descent ambulation mode and LG walking were also recorded for each slope and stair height condition, with the right leg leading the transition. The transition stride was segmented to start at the final right leg toe-off of the previous ambulation mode and end at the first right leg toe-off of the new ambulation mode. Further details about the experimental collection are provided by Camargo *et al.* [33].

Motion capture and 6-axis GRF data were collected using a 36-camera Vicon motion capture system (Oxford Metric, Oxford, UK) and 10 Bertec force plates (Bertec, Columbus, Ohio, USA), respectively. The 200 Hz motion capture data and 1000 Hz GRF data were filtered using a zero-lag, lowpass filter with cutoff frequencies of 6 and 20 Hz, respectively. In addition, wearable sensor data were also collected to be used as input to the hip moment estimation models. Hip flexion/extension angle of the right leg was measured using an electrogoniometer (Biometrics Ltd, UK) at 1000 Hz. Motion capture, GRF, and electrogoniometer data were sampled and synchronized using a Vicon Lock Sync Box. Additionally, accelerometer and gyroscope data were sampled from 6-axis IMUs (Yost, Ohio, USA) mounted on the trunk and right thigh of the participants. IMU data were sampled by and stored on a Raspberry Pi 3 (Raspberry Pi Foundation, Cambridge, UK) at 200 Hz and synchronized with the remaining data by toggling a digital signal directly wired to the Lock Sync Box with each new sample. Unfortunately, the measured IMU data were discarded due to data dropout and signal saturation throughout the experimental data collection and were replaced by synthesized IMU data based on the kinematics of each stride (see Appendix A for implementation details and Supplement I for validation).

B. *OpenSim Hip Moment Labeling*

Ground-truth sagittal hip moment was computed using the opensource, musculoskeletal modeling software, OpenSim v4.1 [34], [35]. The Gait2354 lower limb model was scaled to fit each participant's segment dimensions and inertial properties using the Scale Tool. Lower limb joint kinematics were computed using the Inverse Kinematics Tool, which used weighted least-squares optimization to minimize the squared distance between the measured and modeled marker trajectories as a function of joint position. Lower limb joint moments were then computed for each trial using the Inverse Dynamics Tool, which solved the equations of motion of the scaled anatomical model using the joint kinematics and measured GRFs.

IV. Model Optimization and Evaluation

A. *Biological Hip Moment Estimators*

Four hip moment estimation approaches were implemented in this study: the TCN (our approach) shown in Fig. 2, the non-machine learning baseline method, and two alternative neural networks (i.e., the FCNN and the LSTM) used to benchmark our approach. The output of each estimator was a scalar estimate of the hip flexion/extension moment of the right leg. We considered estimating hip flexion/extension dynamics for this study, since the hip is a primary energy contributor during multimodal ambulation [36] and would require the most wearable sensors to estimate the biological moment using inverse dynamics, compared to the ankle and knee joints.

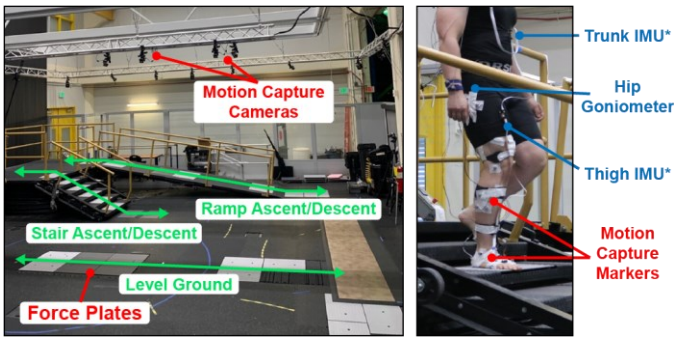

Fig. 1. Experimental setup. Participants walked on level ground, ramps and stairs of several slopes and heights, shown in green. Force plates and a motion capture system were used to collect conventional biomechanics data (red). Wearable sensors were used to collect additional kinematic measurements (blue). *The trunk and thigh inertial measurement unit (IMU) data were replaced with simulated data due to experimental limitations.

The Baseline Method – was included in our study to provide an approach representative of those used by exoskeleton biological torque controllers [8], [9], which provide assistance based on a predefined torque trajectory shaped by normative biomechanical curves. The baseline method was computed as the subject-average hip moment profile with respect to gait phase per ambulation mode and mode transition, thus predicting the average, mode-dependent torque profile for each stride. In our study, the baseline method was informed by a perfectly accurate oracle of both gait phase and ambulation mode. This approach represented the performance of the baseline method in a best-case, but unrealistic, scenario, since gait phase and mode estimates also have error in practice [2]–[5].

Opposed to the baseline method, the neural networks did not rely on gait phase and ambulation mode data. Instead, the inputs to each neural network consisted of the synthesized trunk and thigh 6-axis IMU data, the hip goniometer angle, and the hip angular velocity (14 channels total) (Fig. 1). The hip velocity was computed using first-order backward finite differencing from the hip goniometer angle and was lowpass filtered using a third order Butterworth filter with a cutoff frequency of 10 Hz, which maintained causality of the input data. The ground-truth hip moment computed using OpenSim was used as the label for training and testing the models. Each model was implemented in Python v3.6.9 using the deep learning framework, Pytorch v1.6.0. The hyperparameters of the TCN, FCNN, and LSTM were independently optimized using 16-fold leave-one-subject-out validation (details in Appendix B). Each neural network was trained using the Adam optimizer and an MSE loss function. The optimized models for each type of neural network were then trained for 300 epochs from random initialization using a leave-one-subject-out approach. The results for all

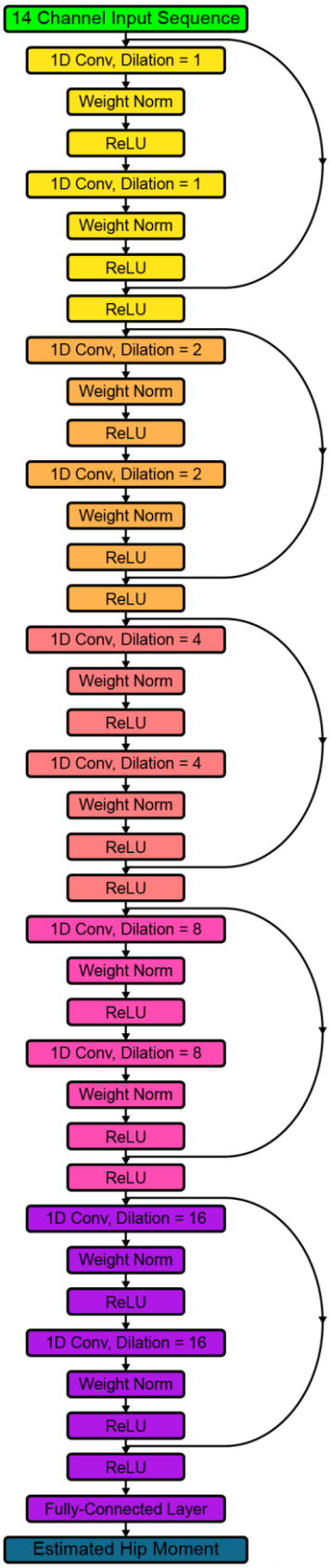

Fig. 2. The hip moment estimation network. The dilation factor is increased after each residual connection, shown with a change in hue, to exponentially increase the receptive field of the network with network depth. Residual connections are used between increases in dilation to further stabilize the network during training.

further analyses were computed using this training method.

The FCNN – served as an additional comparison to our approach, as it has effectively estimated lower limb biological joint moments in previous studies [22], [28]–[30], but has so far diminished in performance when trained on a large set of ambulatory tasks [11]. FCNNs use a series of fully connected layers comprising hidden nodes activated by a nonlinear activation function to estimate the output. Based on our hyperparameter optimization, the FCNN in our study consisted of seven total layers – six hidden layers of size 30, activated by the ReLU function, and an output layer. Batch normalization after each hidden layer was also used to stabilize the network during training. Unique to the FCNN, a feature extraction layer that preceded the fully connected layers was included to reduce the dimensionality and encode temporal relationships in the input data. Specifically, the feature extraction layer computed the mean, standard deviation, and latest value over a sliding window for each input channel, which were the features found to reduce hip moment estimation RMSE in our previous work [29]. This reduced the dimensionality of the FCNN input space from $\mathbb{R}^{1400}$ to $\mathbb{R}^{42}$, given the 100 frame (495 ms) window size of the input data selected as a hyperparameter of the network.

The LSTM – was also used in our study as the final benchmark to our approach. The LSTM is a type of recurrent neural network (RNN), which are a class of neural networks effective in time series regression tasks since they temporally propagate information with each new estimate. This approach allows RNNs to learn latent representations of time history information during backpropagation [37]. LSTMs are particularly useful since they also mitigate the problem of vanishing gradients common to RNNs via the gated structure and cell state within each LSTM node [38]. Since LSTMs model temporal relationships in the data, we did not include a feature extraction layer to the model. Specifically, the input data to the LSTM at a given discrete time instance t was the raw sensor data measured at t, which had dimension $\mathbb{R}^{14}$ and the hidden and cell states output by each LSTM node at t-1. The hidden and cell states were randomly initialized from a normal distribution with zero mean and unit variance. The optimized LSTM was composed of two LSTM layers of size 50 and a fully-connected layer to reshape the network output to size one.

The TCN (Our Approach) – was proposed as our candidate network for estimating biological hip moment (Fig. 2) due to the benefits of dilated, causal convolution for joint moment estimation. Firstly, convolutional layers learn feature representations in the input data, removing the need for hand-engineered methods [39]. Additionally, the fixed window size of 1D causal, convolutional layers have been shown to retain temporal information over longer input sequences in practice compared to LSTMs [12]. These benefits of convolutional networks have yielded impressive performance in a variety of sequence modeling tasks [5], [6], [12], [32]. In this study, we implemented the TCN as introduced by Bai *et al.* [12], which used dilated 1D convolution to exponentially increase the receptive field of the model (h) with a linear increase in its number of layers, which mitigates the total number of learnable parameters of the model compared to a conventional

convolutional network with the same receptive field. Specifically, the j^{th} output of the dilated 1D convolution operation $F(\cdot)$ of a sequence x, is computed as

$$F_j(x) = \sum_{i=0}^{k-1} f_i x_{j-di}\,, \tag{1}$$

where k is the kernel size, f_i is the i^{th} weight of the learned 1D kernel, and d is the dilation factor. From (1), it can be seen that when $d = 1$, the dilated convolution is equivalent to a regular 1D convolutional layer and increasing d increases the range over which the convolution spans the input sequence. From [12], d was exponentially increased with each residual block in the network (each consisting of two convolutional layers), resulting in an exponential increase in h, computed as,

$$h = 1 + \sum_{i=0}^{l-1} 2(k-1)d_i\,, \tag{2}$$

$$d_i = 2^i\,, \tag{3}$$

where l is the number of residual blocks in the network. This approach preserved the resolution of the input sequence. In other words, none of the inputs were "skipped" by the network due to the dilation. Thus, the receptive field of the TCN was determined by the number of layers and kernel size (constant for all layers). The optimized TCN hyperparameters (five residual block levels with channel size of 50 and kernel size of four), resulted in a receptive field of 187 frames (930 ms). Thus, the optimized TCN had an input space of $\mathbb{R}^{14x187}$ for each hip moment estimate in time.

B. Model Generalization to Unseen Gait Contexts

Each neural network variant was tested on the gait transitions collected in the dataset (i.e., RA/RD/SA/SD to LG and LG to RA/RD/SA/SD). Since the training set only included steady-state strides, testing each model on the transition strides evaluated its ability to generalize to a subset of non-cyclic gait. To further test the generalization of the subject-independent TCN, we also tested its performance on multiple hold-out conditions. First, the model was evaluated using a leave-one-ground-slope-out and leave-one-stair-height-out approach, in which the model was trained after random initialization while permuting through the hold-out conditions. With each permutation, the hold-out condition was a single ground slope or stair height, which was then used as the test condition for the subject-independent models. This analysis quantified the performance of the model when generalizing to unseen contexts (i.e., novel ground slopes and stair heights).

C. Statistical Analysis

All statistical tests were completed using SPSS Statistics 21.0 (IBM, Amonk, NY, USA) with an α = 0.05 level of significance. A one-way repeated measures analysis of variance (ANOVA) was used to test for a significant effect among the results of the four hip moment estimation methods during steady-state ambulation and during mode transitions. A two-way repeated measures ANOVA was used to compute main and interaction effects between two hip moment estimators (i.e., the TCN and baseline method) and ambulation modes. Similarly, a two-way repeated measures ANOVA was used to compute significant effects within the ground slope and stair height analyses. A Greenhouse-Geisser correction was used to correct for any violations in sphericity. A *post hoc* multiple comparisons test was used to compute pairwise differences within each comparison with a Bonferroni correction to control the familywise error rate. Finally, a paired t-test was used to compute statistical differences between the overall steady-state and mode transition results of the TCN.

V. Results

A. Steady-State Hip Moment Estimation Performance

The TCN significantly improved the leave-one-subject-out validation RMSE and R^2 compared to the FCNN, LSTM, and baseline method (Table I) ($p<0.05$). Additionally, the FCNN and LSTM models significantly reduced estimation RMSE compared to the baseline method ($p<0.05$) but did not statistically differ in R^2 compared to the baseline method. As shown in Fig. 3, the TCN also reduced the average RMSE during steady-state LG, RA, RD, SA, and SD ambulation by 29.1%, 31.8%, 21.8%, 12.1%, and 25.4%, respectively, compared to the baseline method. These differences were significant for all modes other than SA (Fig. 3) ($p<0.05$). Additionally, the TCN resulted in an average R^2 of 0.904±0.050, 0.935±0.040, 0.895±0.050, 0.897±0.049, and 0.753±0.077 for the LG, RA, RD, SA, and SD modes, respectively. Compared to the baseline method, the TCN significantly increased R^2 for all modes other than RA and SA ($p<0.05$). Fig. 4 shows the estimated hip moment of the TCN and baseline method for representative strides of the steady-state ambulation conditions.

The TCN RMSE did not statistically differ among ambulation modes, but RMSE of the baseline method for RA was significantly higher than that of SA ($p<0.05$). The resulting TCN R^2 for SD was significantly lower than the TCN R^2 for the

TABLE I
HIP MOMENT ESTIMATION RESULTS FOR STEADY-STATE AMBULATION

	Our Approach (TCN)	FCNN	LSTM	Baseline Method
RMSE (Nm/kg)	**0.131±0.018***	0.150±0.024*†	0.149±0.020*†	0.173±0.015†
R^2	**0.880±0.030***	0.848±0.039†	0.842±0.037†	0.793±0.061†

* and † represent a statistical difference from the baseline method and TCN estimates, respectively ($p<0.05$). All other comparisons were not statistically different. Best performance according to RMSE and R^2 is bolded. Results are presented as the leave-one-subject-out average ± 1 standard deviation.

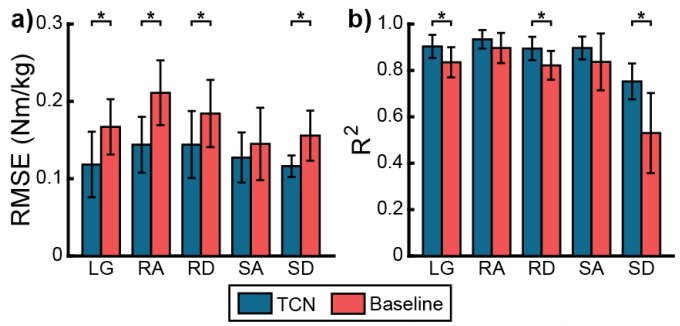

Fig. 3. Hip moment estimation results per ambulation mode. The (a) RMSE (lower is better) and (b) R^2 (higher is better) results of the temporal convolutional network (TCN) and baseline method are shown. Results are presented for the level ground (LG), ramp ascent (RA), ramp descent (RD), stair ascent (SA), and stair descent (SD) ambulation modes. The error bars represent ±1 standard deviation. * represents statistical difference ($p<0.05$). Statistical comparisons among ambulation modes are not shown.

other ambulation modes ($p<0.05$). Similarly, the R^2 of the baseline method was significantly lower for SD compared to the other ambulation modes ($p<0.05$). The difference in baseline R^2 between RA and RD was also significant ($p<0.05$).

B. Model Generalization

As shown in Table II, the TCN significantly improved the leave-one-subject-out validation RMSE and R^2 compared to the alternative approaches when tested on the mode transition dataset ($p<0.05$). Additionally, the LSTM model significantly increased R^2 compared to the baseline method ($p<0.05$), but all other comparisons were not statistically significant. Further, the average RMSE of the TCN estimate on the transition dataset was 0.152±0.027 Nm/kg, which was 16.3% larger than the average steady-state RMSE (Fig. 5a) ($p<0.05$). As shown in Fig. 5b, the average RMSE of the mode transitions ranged from 0.129±0.034 Nm/kg (SA to LG) to 0.186±0.046 Nm/kg (LG to RA). Additionally, the TCN estimate on the transition dataset resulted in an R^2 of 0.786±0.055 (Fig. 5c), which was significantly lower than the R^2 of the steady-state result ($p<0.05$). The average R^2 of each mode transition is shown in Fig. 5d, which ranged from 0.537±0.165 (SD to LG) to 0.918±0.028 (LG to RA).

Fig. 6 shows the TCN results of the ground slope and stair height hold-out analyses. The two-way ANOVA resulted in significant main and interaction effects among the ground slope and hold-out factors when tested on the RMSE and R^2 results ($p<0.05$); however, the 18° hold-out condition was the only condition to significantly reduce model performance according to the multiple comparisons test, increasing the average +18° validation RMSE by 0.077 Nm/kg ($p<0.05$). The remaining leave-one-slope-out comparisons were not statistically significant, resulting in an average increase in RMSE within 0.024 Nm/kg and decrease in R^2 within 0.033, compared to including each hold-out slope in the training set.

When testing the results of the stair height hold-out analysis, the two-way ANOVA resulted in significant main effects among the stair height and hold-out factors ($p<0.05$); however, none of the hold-out conditions resulted in statistical significance during the *post hoc* pairwise comparisons. In general, withholding each stair height increased TCN RMSE within 0.016 Nm/kg and decreased R^2 within 0.034, compared to including each held-out stair height in the training set.

VI. Discussion

This study introduced a novel biological hip moment estimator using a TCN framework, compared its performance to three alternative estimators, and evaluated its generalizability to a variety of overground ambulatory conditions. As hypothesized, the TCN outperformed the FCNN, LSTM, and baseline method when tested on the steady-state ambulation data and mode transition data with respect to RMSE and R^2 (Tables I and II) ($p<0.05$). Additionally, the TCN improved RMSE and R^2 results for each steady-state ambulation mode compared to the baseline method (Fig. 3), significantly improving these outcomes for the LG, RD, and SD ambulation modes ($p<0.05$). Thus, the TCN better captured the changes in hip moment per stride, despite that the baseline method was informed by a perfectly accurate gait phase and ambulation mode oracle. In practice, these estimates also have error, which would further reduce the accuracy of the baseline method [2]–[5]. Therefore, the performance of the TCN is sufficient for modulating exoskeleton assistance, since it outperformed the baseline method, and maintains the potential to generalize across ambulatory tasks.

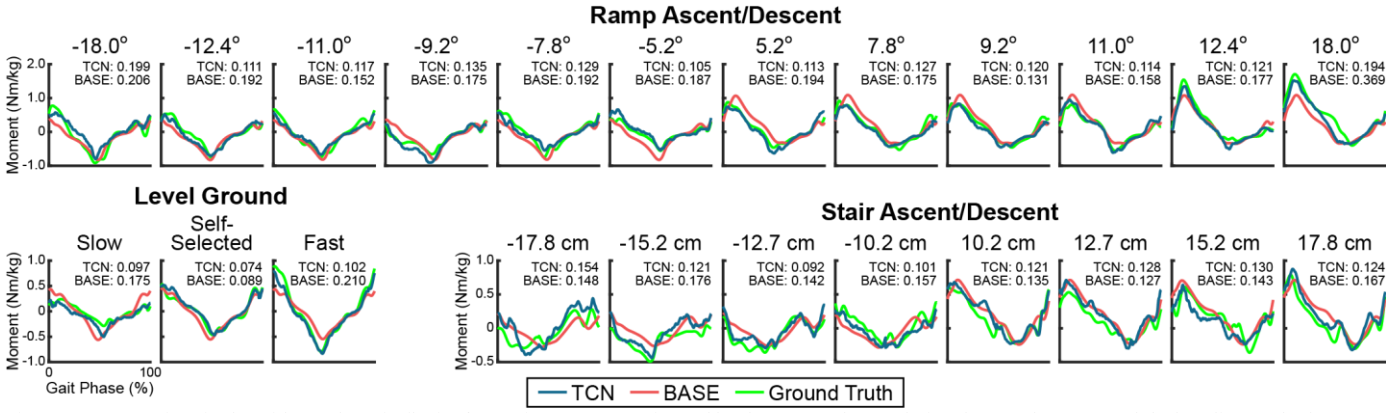

Fig. 4. Representative single stride results. Hip flexion/extension moment estimated by the temporal convolutional network (TCN) and the baseline method (BASE) are shown for representative strides of each ambulation condition. The ground-truth hip moment for each stride is also shown. The resulting RMSE (Nm/kg) of the TCN and BASE method for each selected stride is included for reference. Heel strike denotes the start of the gait cycle.

TABLE II
HIP MOMENT ESTIMATION RESULTS DURING AMBULATION MODE TRANSITIONS

	Our Approach (TCN)	FCNN	LSTM	Baseline Method
RMSE (Nm/kg)	**0.152±0.027***	0.173±0.035†	0.169±0.026†	0.177±0.023†
R^2	**0.786±0.055***	0.738±0.067†	0.757±0.046*†	0.712±0.047†

* and † represent a statistical difference from the baseline method and TCN estimates, respectively ($p<0.05$). All other comparisons were not statistically different. Best performance according to RMSE and R^2 is bolded. Results are presented as the leave-one-subject-out average ± 1 standard deviation.

There were no significant differences in TCN estimation RMSE among the steady-state ambulation modes. We expected this result since the network was trained using the MSE loss function, weighting each ambulation mode with equal importance. However, the SD R^2 (0.753±0.077) was significantly lower than that of all other steady-state ambulation modes, which had an average R^2 of 0.908 ($p<0.05$). Similarly, the resulting R^2 results of the LG to SD and SD to LG transitions were lower than those the other mode transitions (Fig. 5d). The reduction in R^2 of the SD conditions was a result of the decreased range in hip moment along the SD strides compared to the other modes. Therefore, the model maintained performance with respect to the selected loss function; however, it may be important to use a loss function that normalizes loss among ambulation modes if relative estimation performance of each ambulation mode is more important than absolute error. For instance, normalizing loss across ambulation modes may be an important consideration when developing a biological moment estimator to be used for multimodal, out-of-lab biomechanical analyses; however, it may be more appropriate to minimize absolute error when developing a similar system for exoskeleton control.

The TCN was robust to 19 of the 20 ground slope and stair height hold-out conditions (Fig. 6), resulting in a maximum increase in RMSE within 0.024 Nm/kg and decrease in R^2 within 0.034, other than for the +18° hold-out condition. The reduced performance on the +18° hold-out was expected since this condition was 45% steeper than the second most severe ground slope condition of 12.4°. Therefore, it is important to prioritize collecting training data at steep inclines if these conditions are within the test set distribution (e.g., if using a neural network to estimate joint moments during hiking); however, a reduced set of ramp conditions can be used in the training dataset if this is less of a concern (e.g., for joint moment estimation during typical community ambulation), since the model generalized well when holding out less severe ground slopes. Additionally, the TCN was robust against the stair

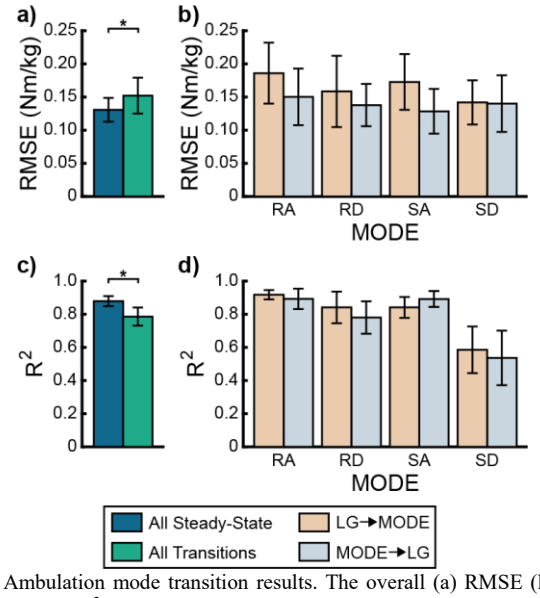

Fig. 5. Ambulation mode transition results. The overall (a) RMSE (lower is better) and (c) R^2 (higher is better) results of the temporal convolutional network (TCN) are shown for the steady-state and ambulation mode transition data. Additionally, the TCN (b) RMSE and (d) R^2 results of each ambulation mode transition involving level ground (LG), ramp ascent (RA), ramp descent (RD), stair ascent (SA), and stair descent (SD) are shown. The error bars represent ±1 standard deviation. * represents statistical difference ($p<0.05$).

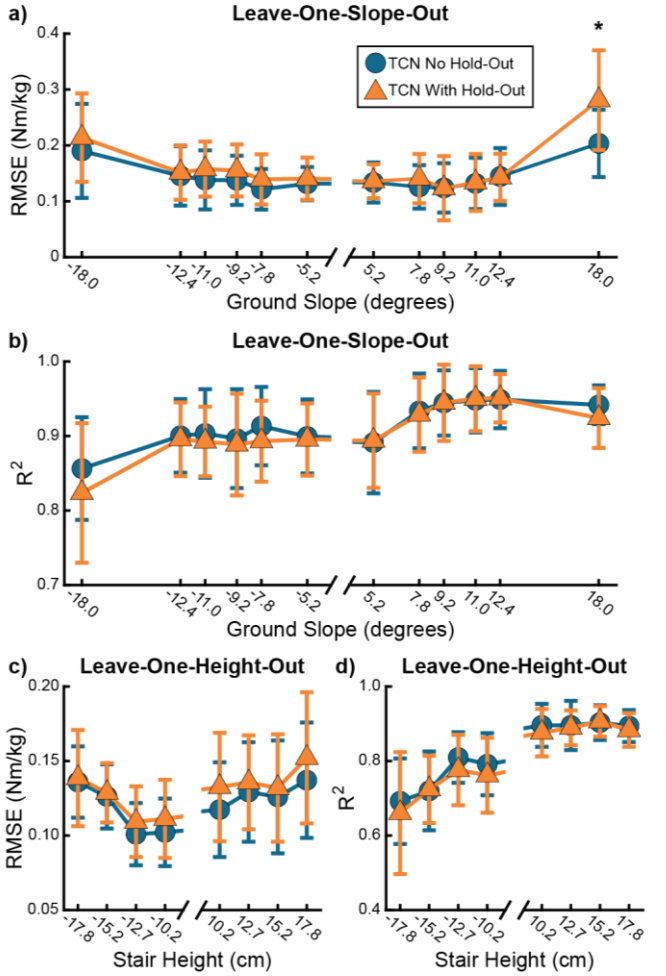

Fig. 6. Leave-out analysis results. The (a) RMSE (lower is better) and (b) R^2 (higher is better) results of the temporal convolutional network (TCN) are shown with and without withholding each ground slope condition from the training set. The (c) RMSE and (d) R^2 results of the TCN with and without withholding each stair height from the training dataset are also shown. The error bars represent ±1 standard deviation. * represents statistical difference comparing the results when each ground slope and stair height was and was not included in the training dataset ($p<0.05$).

height hold-out conditions, which encompassed the complete range of ADA compliant stair heights. Therefore, it is unlikely that additional stair height conditions would need to be included in the training dataset to ensure the model can generalize to stair ascent/descent; however, additional SD data and improved model training methods may improve estimation R^2 on the SD trials in general.

While the TCN model performed significantly worse on the ambulation mode transition data compared to the overall steady-state results, as shown in Fig. 5a and 5c ($p<0.05$), the transition data results were of similar magnitude to the steady-state results of the baseline method. This comparable result suggests that the mode transition performance may be sufficient for exoskeleton control, since exoskeleton controllers similar to the baseline method have been shown to augment human walking [8], [9]. In future work, including transition data in the training set will likely improve these results.

Though challenging to compare due to experimental differences, the LG validation performance of the TCN in our study was competitive with those of previous subject-independent joint moment estimators that used wearable sensors. The TCN in our study resulted in an average RMSE and R^2 of 0.118±0.042 Nm/kg and 0.904±0.050, respectively, for steady-state LG walking. Using an IMU-based CNN trained on LG walking and running data, Dorschky *et al.* reported a hip moment estimation RMSE and correlation coefficient of 0.17 Nm/kg and 0.91 (approximate R^2 of 0.82), respectively, during LG walking [32]. Similarly, Mundt *et al.* reported a sagittal hip moment correlation coefficient above 0.95 (approximate R^2 above 0.90) during the stance phase of LG walking; however, their IMU-based FCNN model was noncausal and only trained on LG walking data [30]. Additionally, the LG results of our TCN were comparable to those of inverse dynamics-based models that require additional wearable sensor input. For instance, Forner-Cordero *et al.* reported a sagittal hip moment RMSE of 0.15 Nm/kg and correlation coefficient of 0.92 (approximate R^2 of 0.85), using a linked-segment model and pressure insoles [13]. Therefore, the TCN model of our study maintained LG performance compared to previous literature, while proving to generalize to additional common ambulation modes. Additionally, we found that model performance was maintained despite the substantially reduced sensor suite, which is encouraging for future implementations of this joint moment estimation method, especially when onboard an exoskeleton. Since joint moment estimation of a novel subject during ramp and stair walking and during mode transitions has not been previously characterized, our results can stand as a benchmark for following joint moment estimation studies.

This study contained multiple limitations. Due to experimental difficulties, we substituted the experimentally collected thigh and trunk IMU data with synthesized IMU data based on the OpenSim anatomical model and stride kinematics for each participant and trial (implementation details in Appendix A and validation in Supplement I). This approach assumed that the IMUs were placed at the same location and orientation among participants and that the IMU was rigidly connected to the skeletal system of the participant. In practice, it is likely that soft tissue deformation, changes in subject anthropometry, and sensor noise would induce additional complexity in the IMU data that was not included in our dataset. Another limitation of this study is that the entire training dataset was composed of steady-state ambulation data. It is likely that including additional mode transition data in the training dataset will improve model generalizability to additional ambulatory tasks; however, this was outside the scope of our current study. Finally, the neural networks implemented in this study required data from a goniometer, thigh-mounted IMU, and trunk-mounted IMU. Though this is a much simpler sensor suite than those of alternative approaches (e.g., requiring complete kinematic and kinetic measurements of the distal joints or EMG measurements), currently existing technology (e.g., hip exoskeletons) may not measure sagittal hip kinematics and thigh and trunk IMU data by default. In this case, an analysis of the importance of each input sensor could be used to reduce the required sensor suite or wearable devices would need to be updated with additional sensors.

VII. Conclusion

This study proposed a novel, subject-independent biological hip moment estimator using the TCN framework and a limited set of kinematic wearable sensors localized around the hip joint. Our approach resulted in an average estimation RMSE of 0.131±0.018 Nm/kg and R^2 of 0.880±0.030 when tested on multimodal, overground walking. These results outperformed the FCNN and LSTM networks as well as the baseline method, which was representative of methods used by previous exoskeleton controllers [8], [9] ($p<0.05$). Not only did our model result in competitive performance with previous machine learning and inverse dynamics-based hip moment estimators in the literature, but the TCN also generalized well to a variety of ambulatory tasks and hold-out conditions. In general, the TCN was robust to withholding each slope and stair height from the training set, except for when tested on the steepest (+18°) slope ($p<0.05$). Additionally, the TCN RMSE and R^2 was 0.152±0.027 Nm/kg and 0.786±0.055, respectively, when tested on the ambulation mode transitions, which was worse than the steady-state ambulation results ($p<0.05$); however, the transition results were similar to the steady-state results of the baseline method, suggesting the TCN performance is suitable for exoskeleton control. In the future, this system should be validated in real-time for biomechanical analyses, health monitoring, and exoskeleton control.

VIII. Appendix

A. IMU Synthetization

Due to saturation and dropout of the thigh and trunk IMUs, the accelerometer and gyroscope data were replaced using synthesized data from the OpenSim model during each recorded stride. The synthesized data was computed for each trial using the OpenSim API. First, the homogenous transformation matrix ${}^{G}T^{B}$ from the reference frame of a given model body B to the ground inertial reference frame G was computed for the trunk and right femur as,

$$ {}^{G}T^{B} = \begin{bmatrix} {}^{G}R^{B} & p_{GB} \\ \vec{0} & 1 \end{bmatrix}, \tag{4} $$

where ${}^{G}R^{B}$ is the rotation matrix from B to G and p_{GB} is the vector from G to B. At each timestep, ${}^{G}T^{B}$ was computed using the getTransformInGround() method of the OpenSim::Body class after updating the model coordinates based on the inverse kinematics results.

To synthesize the gyroscopic data, the angular velocity vector ${}^{G}\omega^{B}$ of B in G was computed using the angular velocity tensor W as,

$$ {}^{G}\omega^{B} = \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix}, \tag{5} $$

$$ W = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} = \frac{d{}^{G}R^{B}}{dt} \cdot {}^{B}R^{G}, \tag{6} $$

$$ {}^{B}R^{G} = {}^{G}R^{B^T}. \tag{7} $$

The gyroscopic data ${}^{B}\omega^{B}$ was then computed as the angular velocity vector in the local frame of the body using,

$$ {}^{B}\omega^{B} = {}^{B}R^{G} \cdot {}^{G}\omega^{B}. \tag{8} $$

Since the IMUs were not located at the origin of the bodies (i.e., the trunk and right femur), an additional vector p_{BC} was introduced to approximate the location of each IMU with respect to its corresponding body, where C represents the IMU reference frame. Fig. 7 shows the body and IMU reference frames of the trunk and right femur used in our study, and Table III shows the locations of each IMU described by its corresponding reference frame. The vector from G to C was then computed as,

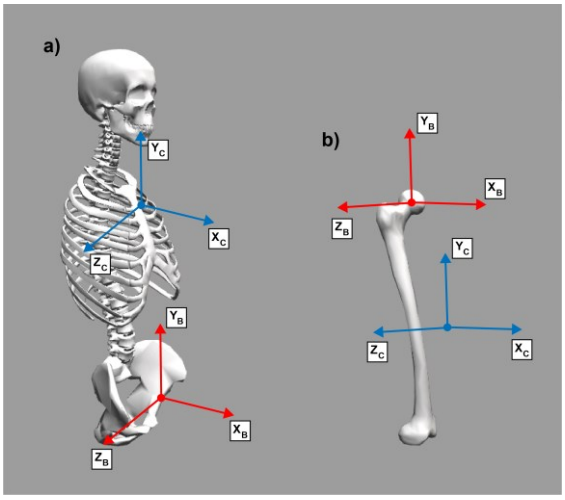

Fig. 7. Synthesized IMU locations. The reference frames of the (a) trunk and (b) thigh are shown in red. Additionally, the reference frames of the synthesized trunk and thigh IMUs are shown in blue.

TABLE III
SYNTHESIZED IMU LOCATIONS

Model Body	X-Axis Translation (mm)	Y-Axis Translation (mm)	Z-Axis Translation (mm)
Trunk	-52	415	0
Right Thigh	76	-206	12

Translations are expressed in terms of the reference frame of the model body.

$$ p_{GC} = p_{GB} + {}^{G}R^{B} \cdot p_{BC}. \tag{9} $$

The synthesized accelerometer data expressed in G was then computed as,

$$ a_{GC} = -\frac{d^2 p_{GC}}{dt^2} + {}^{G}g, \tag{10} $$

where ${}^{G}g$ is the gravity vector in G. Finally, the accelerometer data was then transformed into its local frame as,

$$ a_C = {}^{C}R^{G} \cdot a_{GC}, \tag{11} $$

where

$$ {}^{C}R^{G} = {}^{B}R^{G}, \tag{12} $$

since B and C are located on the same rigid body. The synthesized accelerometer data was then converted from m/s^2 to g-force by dividing a_C by the acceleration of gravity.

B. Neural Network Hyperparameter Optimization

To ensure a fair comparison between TCN, FCNN, and LSTM, the hyperparameters of each neural network variant were optimized using 16-fold leave-one-subject-out validation. Specifically, sixteen neural network models (one per subject) were trained per set of hyperparameters, while the validation subject data was withheld from the training set. The candidate values for each hyperparameter were decided based on preliminary sweeps of the hyperparameters during pilot testing. Early stopping with a patience of 50 epochs and stopping criteria based on validation loss was used to prevent overfitting during the hyperparameter sweep. Each model was trained using the Adam optimizer and the mean-square-error loss function. The average RMSE among the sixteen models was used to score each set of hyperparameters. The final set of hyperparameters for each type of neural network was selected as the set that scored within 1% of the best validation RMSE among all hyperparameter combinations and that minimized the number of learnable parameters of the network. The number of learnable parameters was computed as the total number of elements among network parameters with a true requires_grad flag in Pytorch v1.6.0. This approach maintained model performance while minimizing training time and the likelihood of overfitting to the training set distribution.

1) TCN Hyperparameter Optimization

The number of levels (two 1D convolutional layers per level), channel size, kernel size, dropout probability, and learning rate of the TCN were optimized (Table IV). To maintain a tractable number of hyperparameter combinations, the hyperparameter optimization was separated into two steps. During the first step, the number of levels, number of channels per hidden layer, and kernel size were optimized using a dropout probability of 0.2 and learning rate of 0.0001, which were selected from pilot results. Additionally, combinations of hyperparameters that resulted in a model receptive field above 1000 ms were omitted since the motion capture data used to synthesize the IMUs were often unreliable outside of these bounds due to limitations of the capture space. The best model from this sweep resulted in a validation RMSE of 0.132±0.016 Nm/kg using 237,761 learnable parameters; however, the selected model resulted in a validation RMSE of 0.133±0.017 Nm/kg (within 1% of best model RMSE) using 94,601 learnable parameters. Using the number of levels, channel size, and kernel size of the selected model, the dropout probability and learning rate were then optimized during the second step of the hyperparameter optimization by sweeping the candidate values shown in Table IV. The finalized TCN model resulted in a validation RMSE of 0.131±0.016 Nm/kg.

2) FCNN Hyperparameter Optimization

The number of layers, hidden size, dropout probability, input window size, and learning rate of the FCNN were optimized (Table V). Additionally, batch normalization after the hidden layers of the model was also tested for each combination of hyperparameters. Based on pilot results, the initial window size and learning rate were fixed to 500 ms and 0.0001, respectively, to maintain a tractable number of hyperparameter combinations. Using these values, each combination of the remaining hyperparameter candidate values was evaluated. The best model from this sweep resulted in a validation RMSE of 0.148±0.016 Nm/kg using 35,501 learnable parameters; however, the selected model resulted in a validation RMSE of 0.149±0.018 Nm/kg (within 1% of best model RMSE) using 6,331 learnable parameters. Using the number of layers, hidden size, and dropout probability of the selected model, the final window size and learning rate were then determined by sweeping the candidate values shown in Table V. The finalized FCNN model resulted in a validation RMSE of 0.146±0.018 Nm/kg using the hyperparameters shown in Table V.

TABLE IV
TCN HYPERPARAMETER OPTIMIZATION

Hyperparameter	Candidate Values	Selected Value
# of Levels	1, 2, 3, 4, 5	5
# of Channels per Hidden Layer	10, 30, 50, 80	50
Kernel Size	4, 5, 7, 10, 20	4
Dropout Probability[a]	0, 0.1, 0.2, 0.3, 0.4, 0.5	0.3
Learning Rate[a]	1e-4, 5e-4, 1e-3	5e-4
# of Learnable Parameters	N/A	94,601

[a]The dropout probability and learning rate were tuned during the second optimization step after tuning the remaining hyperparameters.

TABLE V
FCNN HYPERPARAMETER OPTIMIZATION

Hyperparameter	Candidate Values	Selected Value
# of Layers	1, 2, 5, 7, 10	7
# of Nodes per Hidden Layer	10, 30, 50, 80, 100	30
Batch Normalization	True, False	True
Dropout Probability	0, 0.2, 0.4, 0.6	0
Window Size (frames)[a]	10, 20, 50, 80, 100, 120, 150, 180, 200	100
Learning Rate[a]	5e-5, 1e-4, 2e-4, 5e-4, 1e-3, 5e-3	5e-4
# of Learnable Parameters	N/A	6,331

[a]The window size and learning rate were tuned during the second optimization step after tuning the remaining hyperparameters.

3) LSTM Hyperparameter Optimization

The number of layers, number of LSTM cells per hidden layer, dropout probability, and learning rate of the LSTM network were optimized (Table VI). To maintain a tractable number of hyperparameter combinations, the hyperparameter optimization was separated into two steps. During the first step, the number of layers, number of cells per hidden layer, and dropout probability were optimized using a learning rate of 0.0001, which was selected from pilot results. The best model from this sweep resulted in a validation RMSE of 0.150±0.021 Nm/kg using 33,651 learnable parameters. This model architecture also had the lowest number of learnable parameters compared to all other models that resulted in a validation RMSE within 1% of the best model. Using the number of layers, number of cells per layer, and dropout probability of the selected model, the learning rate was then optimized during the second step of the hyperparameter optimization by sweeping the candidate values shown in Table VI. The finalized LSTM model resulted in a validation RMSE of 0.149±0.018 Nm/kg.

REFERENCES

[1] D. A. Winter, *Biomechanics and motor control of human movement*. John Wiley & Sons, 2009.

[2] J. Jang, K. Kim, J. Lee, B. Lim, and Y. Shim, "Online gait task recognition algorithm for hip exoskeleton," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 5327–5332.

[3] K. Yuan *et al.*, "A realtime locomotion mode recognition method for an active pelvis orthosis," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 6196–6201.

[4] D. Quintero, D. J. Lambert, D. J. Villarreal, and R. D. Gregg, "Real-time continuous gait phase and speed estimation from a single sensor," in *2017 IEEE Conference on Control Technology and Applications (CCTA)*, 2017, pp. 847–852.

TABLE VI
LSTM HYPERPARAMETER OPTIMIZATION

Hyperparameter	Candidate Values	Selected Value
# of Layers	2, 3, 4	2
# of LSTM Cells per Hidden Layer	10, 30, 50, 80, 100	50
Dropout Probability	0, 0.1, 0.2, 0.3, 0.4	0.2
Learning Rate[a]	1e-4, 5e-4, 1e-3	5e-4
# of Learnable Parameters	N/A	33,651

[a]The learning rate was tuned during the second optimization step after tuning the remaining hyperparameters.

[5] I. Kang, D. Molinaro, S. Duggal, Y. Chen, P. Kunapuli, and A. Young, "Real-time gait phase estimation for robotic hip exoskeleton control during multimodal locomotion," *IEEE Robot. Autom. Lett.*, 2021.
[6] D. Lee, I. Kang, D. Molinaro, A. Yu, and A. Young, "Real-Time User-Independent Slope Prediction using Deep Learning for Modulation of Robotic Knee Exoskeleton Assistance," *IEEE Robot. Autom. Lett.*, 2021.
[7] G. S. Sawicki, O. N. Beck, I. Kang, and A. J. Young, "The exoskeleton expansion: improving walking and running economy," *J. NeuroEngineering Rehabil.*, vol. 17, no. 1, pp. 1–9, 2020.
[8] T. Lenzi, M. C. Carrozza, and S. K. Agrawal, "Powered hip exoskeletons can reduce the user's hip and ankle muscle activations during walking," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 21, no. 6, pp. 938–948, 2013.
[9] K. Seo, J. Lee, Y. Lee, T. Ha, and Y. Shim, "Fully autonomous hip exoskeleton saves metabolic cost of walking," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 4628–4635.
[10] J. Zhang *et al.*, "Human-in-the-loop optimization of exoskeleton assistance during walking," *Science*, vol. 356, no. 6344, pp. 1280–1284, 2017.
[11] B. J. Stetter, F. C. Krafft, S. Ringhof, T. Stein, and S. Sell, "A machine learning and wearable sensor based approach to estimate external knee flexion and adduction moments during various locomotion tasks," *Front. Bioeng. Biotechnol.*, vol. 8, 2020.
[12] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *ArXiv Prepr. ArXiv180301271*, 2018.
[13] A. Forner-Cordero, H. J. F. M. Koopman, and F. C. T. van der Helm, "Inverse dynamics calculations during gait with restricted ground reaction force information from pressure insoles," *Gait Posture*, vol. 23, no. 2, pp. 189–199, Feb. 2006, doi: 10.1016/j.gaitpost.2005.02.002.
[14] L. Ren, R. K. Jones, and D. Howard, "Whole body inverse dynamics over a complete gait cycle based only on measured kinematics," *J. Biomech.*, vol. 41, no. 12, pp. 2750–2759, 2008.
[15] T. Khurelbaatar, K. Kim, S. Lee, and Y. H. Kim, "Consistent accuracy in whole-body joint kinetics during gait using wearable inertial motion sensors and in-shoe pressure sensors," *Gait Posture*, vol. 42, no. 1, pp. 65–69, 2015.
[16] S. Lee, K. Kim, Y. H. Kim, and S. Lee, "Motion anlaysis in lower extremity joints during ski carving turns using wearble inertial sensors and plantar pressure sensors," in *2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2017, pp. 695–698.
[17] A. Karatsidis *et al.*, "Musculoskeletal model-based inverse dynamic analysis under ambulatory conditions using inertial motion capture," *Med. Eng. Phys.*, vol. 65, pp. 68–77, 2019.
[18] S. Skals, M. K. Jung, M. Damsgaard, and M. S. Andersen, "Prediction of ground reaction forces and moments during sports-related movements," *Multibody Syst. Dyn.*, vol. 39, no. 3, pp. 175–195, 2017.
[19] K. Kanjanapas and M. Tomizuka, "7 Degrees of Freedom Passive Exoskeleton for Human Gait Analysis: Human Joint Motion Sensing and Torque Estimation During Walking," *IFAC Proc. Vol.*, vol. 46, no. 5, pp. 285–292, 2013.
[20] A. Forner Cordero, H. J. F. M. Koopman, and F. C. T. van der Helm, "Use of pressure insoles to calculate the complete ground reaction forces," *J. Biomech.*, vol. 37, no. 9, pp. 1427–1432, Sep. 2004, doi: 10.1016/j.jbiomech.2003.12.016.
[21] A. M. Howell, T. Kobayashi, H. A. Hayes, K. B. Foreman, and S. J. M. Bamberg, "Kinetic Gait Analysis Using a Low-Cost Insole," *IEEE Trans. Biomed. Eng.*, vol. 60, no. 12, pp. 3284–3290, Dec. 2013, doi: 10.1109/TBME.2013.2250972.
[22] D. A. Jacobs and D. P. Ferris, "Estimation of ground reaction forces and ankle moment with multiple, low-cost sensors," *J. NeuroEngineering Rehabil.*, vol. 12, no. 1, p. 90, Dec. 2015, doi: 10.1186/s12984-015-0081-x.
[23] S. E. Oh, A. Choi, and J. H. Mun, "Prediction of ground reaction forces during gait based on kinematics and a neural network model," *J. Biomech.*, vol. 46, no. 14, pp. 2372–2380, 2013.
[24] F. J. Wouda *et al.*, "Estimation of Vertical Ground Reaction Forces and Sagittal Knee Kinematics During Running Using Three Inertial Sensors," *Front. Physiol.*, vol. 9, 2018, doi: 10.3389/fphys.2018.00218.
[25] D. G. Lloyd and T. F. Besier, "An EMG-driven musculoskeletal model to estimate muscle forces and knee joint moments in vivo," *J. Biomech.*, vol. 36, no. 6, pp. 765–776, 2003.
[26] T. S. Buchanan, D. G. Lloyd, K. Manal, and T. F. Besier, "Neuromusculoskeletal modeling: estimation of muscle forces and joint moments and movements from measurements of neural command," *J. Appl. Biomech.*, vol. 20, no. 4, pp. 367–395, 2004.
[27] G. Durandau, D. Farina, and M. Sartori, "Robust real-time musculoskeletal modeling driven by electromyograms," *IEEE Trans. Biomed. Eng.*, vol. 65, no. 3, pp. 556–564, 2017.
[28] M. E. Hahn and K. B. O'Keefe, "A neural network model for estimation of net joint moments during normal gait," *J. Musculoskelet. Res.*, vol. 11, no. 03, pp. 117–126, 2008.
[29] D. D. Molinaro, I. Kang, J. Camargo, and A. J. Young, "Biological Hip Torque Estimation using a Robotic Hip Exoskeleton," in *2020 8th IEEE RAS/EMBS International Conference for Biomedical Robotics and Biomechatronics (BioRob)*, 2020, pp. 791–796.
[30] M. Mundt *et al.*, "Estimation of Gait Mechanics Based on Simulated and Measured IMU Data Using an Artificial Neural Network," *Front. Bioeng. Biotechnol.*, vol. 8, 2020.
[31] M. M. Ardestani *et al.*, "Human lower extremity joint moment prediction: A wavelet neural network approach," *Expert Syst. Appl.*, vol. 41, no. 9, pp. 4422–4433, 2014.
[32] E. Dorschky, M. Nitschke, C. F. Martindale, A. J. van den Bogert, A. D. Koelewijn, and B. M. Eskofier, "CNN-Based Estimation of Sagittal Plane Walking and Running Biomechanics From Measured and Simulated Inertial Sensor Data," *Front. Bioeng. Biotechnol.*, vol. 8, p. 604, 2020.
[33] J. Camargo, A. Ramanathan, W. Flanagan, and A. Young, "A comprehensive, open-source dataset of lower limb biomechanics in multiple conditions of stairs, ramps, and level-ground ambulation and transitions," *J. Biomech.*, vol. 119, p. 110320, 2021.
[34] S. L. Delp *et al.*, "OpenSim: open-source software to create and analyze dynamic simulations of movement," *IEEE Trans. Biomed. Eng.*, vol. 54, no. 11, pp. 1940–1950, 2007.
[35] A. Seth *et al.*, "OpenSim: Simulating musculoskeletal dynamics and neuromuscular control to study human and animal movement," *PLoS Comput. Biol.*, vol. 14, no. 7, p. e1006223, 2018.
[36] R. W. Nuckols, K. Z. Takahashi, D. J. Farris, S. Mizrachi, R. Riemer, and G. S. Sawicki, "Mechanics of walking and running up and downhill: A joint-level perspective to guide design of lower-limb exoskeletons," *PLoS One*, vol. 15, no. 8, p. e0231996, 2020.
[37] P. J. Werbos, "Backpropagation through time: what it does and how to do it," *Proc. IEEE*, vol. 78, no. 10, pp. 1550–1560, 1990.
[38] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[39] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.