Lower Limb Torque Prediction for Sit-To-Walk Strategies Using Long Short-Term Memory Neural Networks

Chamalka Kenneth Perera, *Graduate Student Member, IEEE*, Alpha. A. Gopalai, *Member, IEEE*, Darwin Gouwanda, *Member, IEEE*, Siti. A. Ahmad, *Senior Member, IEEE*, and Pei-Lee Teh, *Member, IEEE*

***Abstract*— Joint torque prediction is crucial when investigating biomechanics, evaluating treatments, and designing powered assistive devices. Controllers in assistive technology require reference torque trajectories to set the level of assistance for a patient during rehabilitation or when aiding essential daily tasks such as sit-to-walk (STW). STW itself can be generalized into strategies based on individual needs and movement patterns. In this study, three long short-term memory (LSTM) neural networks were empirically trained for hip and knee torque prediction considering these STW strategies and subject anthropometry. The hip and knee are the drivers of STW, while the network architectures were selected for recognizing temporal and spatial relationships. Performance of the LSTMs were compared and evaluated against the STW strategies to accurately generate strategy-specific and user-oriented torque. As such, train and test STW data were obtained from 65 subjects across three age groups: young, middle-aged, and older adults (19-73 years). Model inputs were hip and knee angles with horizontal center of mass velocity, while windowing allowed the LSTMs to dynamically adapt to real-time changes in STW transitions. The encoder-decoder LSTM showcased optimal performance with robust recognition of temporal features. It produced significantly (P <0.05) low hip and knee root mean square error (0.24 ± 0.07 and 0.15 ± 0.02 Nm/kg), strong Spearman's correlation (93.43 ± 2.86 and 84.83 ± 2.96%) and good intraclass correlation coefficients (greater than 0.75), demonstrating model reliability. Hence, this network predicts strategy and user oriented reference torques for personalized controllers in assistive devices, with more natural application of assistance.**

***Index Terms*— Encoder-decoder, CNN-LSTM, strategy classification, torque controllers, assistive devices.**

Received 24 May 2024; revised 18 August 2024 and 23 September 2024; accepted 22 October 2024. Date of publication 30 October 2024; date of current version 6 November 2024. This work was supported by the Ministry of Higher Education, Malaysia, under Project FRGS/1/2022/TK07/MUSM/02/2. *(Corresponding author: Alpha. A. Gopalai.)*

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the Monash University Human Research Ethics Committee under Project No. 32328.

Chamalka Kenneth Perera, Alpha. A. Gopalai, and Darwin Gouwanda are with the School of Engineering, Monash University, Subang Jaya, Selangor 47500, Malaysia (e-mail: chamalka.perera@monash.edu; alpha.agape@monash.edu; darwin.gouwanda@monash.edu).

Siti. A. Ahmad is with the Faculty of Engineering, Universiti Putra Malaysia, Selangor 43400, Malaysia (e-mail: sanom@upm.edu.my).

Pei-Lee Teh is with the School of Business, Monash University, Subang Jaya, Selangor 47500, Malaysia (e-mail: teh.pei.lee@monash.edu).

Digital Object Identifier 10.1109/TNSRE.2024.3488052

I. Introduction

ACTIVE assistive devices are governed by torque controllers which function on accurate joint torque prediction. These predicted reference torque trajectories from healthy adults set a benchmark to gauge the level of assistance a patient requires and ensures sufficient assistive torque is generated to execute a motion correctly [1], [2]. Torque profiles are also essential when studying movement biomechanics, investigating impaired motion, and evaluating the efficacy of interventions or treatments [3], [4], [5]. Advancements in torque prediction methods would significantly benefit the design and control architecture of assistive devices (for example exoskeletons or soft robotic exosuits), in rehabilitation or when aiding daily activities due to movement impairment [6], [7].

Currently, multiple methods exist for deriving reference joint torque. These include inverse dynamics (ID), mathematical modelling, neuromusculoskeletal modelling, and neural networks. ID provides ground truth and gold standard torque trajectories using motion capture (Mocap) with body kinematic models, employing Newton-Euler equations of motion. However, expensive and bulky Mocap equipment is required with tedious and time consuming data collection and processing that limits ease of integration with wearable devices [7], [8]. Mathematical modelling produces quick and robust torque estimates using the joint torque-angle relationship with methods such as gravity compensation or Euler-Lagrange. Existing assistive devices use these models due to their ease of implementation and low computational requirements [1], [9], [10], [11]. However, they target only specific motions (for example standing or gait) and produce generic torque profiles that are not tailored to individual characteristics, anthropometry, or movement strategies. Mathematical models may also produce large torque

estimation errors leading to high controller gains, compensation factors, and unstable control [12]. Neuromusculoskeletal modelling uses surface electromyography (SEMG) to derive muscle force through activation and contraction dynamics, which when combined with the muscle moment-arms produces joint torque [13]. This type of torque estimation can be performed using biomechanical modelling software like OpenSim [14], and is commonly utilized for muscle or joint analysis, producing subject-specific torque profiles. Yet, it requires laborious calibration and processing while SEMG signals are noisy, affected by sweat, sensor placement, and is not ideal for prolonged use in wearable devices [3], [5]. Alternatively, neural networks map directly from sensor input to output torque by capturing temporal and spatial relationships. They can produce assistive torque based on individual anthropometry, characteristics, and executions. To achieve this, large training datasets that cover all aspects of a subject population are required [7], [15].

Sit-to-walk (STW) torque prediction is vital for lower limb assistive devices, as this simple but essential weight-bearing transition directly contributes to an individual's mobility, independence and thus, quality of life [4]. STW is defined as a fluid merging of sit-to-stand with gait [16] and is classified by three generalized movement strategies: (1) forward continuation, (2) balance, and (3) sit-to-stand-and-walk (STSW) [4]. Each strategy is defined by varying biomechanics and specific torque profiles as detailed in our previous work [4]. Additionally, the hip and knee extensor muscles are the primary drivers to this motion [17]. Generating and predicting STW strategy and user specific hip and knee torque would result in personalized controllers for assistive devices with more natural delivery of assistance.

For torque controllers, neural networks could overcome the limitations of traditional methods (ID, mathematical and neuromusculoskeletal modelling) by removing the need for bulky equipment, tedious data processing, and human biomechanical models while still providing tailored and strategy-specific assistance [3], [7]. Neural models for STW torque prediction require timeseries sequence-to-sequence mapping. Long short-term memory (LSTM) architectures demonstrated strong performance in this aspect, for predicting joint kinematics, kinetics, and torques in literature [5], [7], [18], [19], [20]. Therefore, this architecture was considered using hip and knee joint angles with horizontal center of mass (HCOM) velocity as inputs. These quantities were selected as they significantly vary with an individual's chosen strategy [4], are easily measured, and integrated into wearable technology.

This study aims to develop LSTM models that predict hip and knee flexion/extension reference torques, accounting for STW strategies and subject anthropometry by normalization with bodyweight. The LSTMs are evaluated across all three strategies to ensure they meet the strategy-specific torque requirements and are representative of all STW executions [4]. Three LSTM models were trained: (1) Vanilla LSTM as the base model, (2) Encoder-Decoder LSTM for temporal feature extraction, and (3) Convolution Neural Network (CNN)–LSTM for spatial features. The LSTMs utilized input and output sliding windows for real-time operation

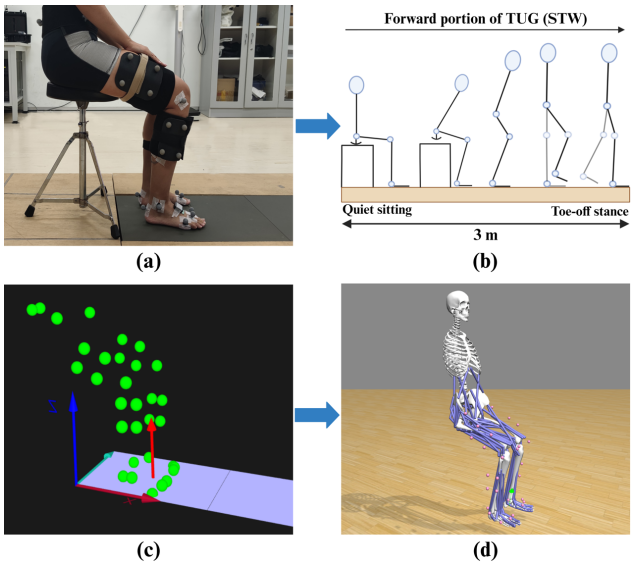

Fig. 1. Sit-to-walk (STW) motion capture and biomechanical analysis, where (a) shows the initialized sitting position and (b) illustrates the timed-up-and-go (TUG) test, from which STW was obtained between quiet-sitting and toe-off of the stance foot. (c) The motion capture markers and ground reaction forces were processed using Qualisys Track Manager (Qualisys, Sweden). This was used in (d) to derive joint angles, torques, and horizontal center of mass (HCOM) velocity using the scaled Gait 2392 musculoskeletal model in OpenSim.

mirroring torque controllers. The subsequent sections discuss the experimental procedure, dataset preparation, LSTM architectures, and STW strategy-wise performance, culminating in discussions for the optimal model.

II. Methods

A. Experimental Details

STW lower body Mocap and force plate data were collected from 65 subjects ranging from 19-73 years and spanning three age groups based on chronological ageing, as introduced in reference [21]. These groupings were young (19-35 years; n = 32; mean weight of 64.12 ± 17.04 kg), middle-aged (36-55 years; n = 15; mean weight of 67.94 ± 15.85 kg) and older adults (56-73 years; n = 18; mean weight of 59.38 ± 10.7 kg), with the total subject group having a mean age of 39.35 ± 17.51 years. The dataset comprised 325 trials inclusive of all three STW strategies and the distribution by biological sex included 32 males and 33 females. For data collection a Qualisys (Sweden) Mocap system sampling at 200 Hz and three Bertec (USA) force plates sampling at 1 kHz were used [22]. Accordingly, for data processing, training, and testing of the LSTM models, Python version 3.9.18, TensorFlow version 2.10.1 and a computer with a 11th Gen Intel® Core™ i7-11800H CPU @ 2.30 GHz, NVIDIA GeForce RTX 3050 GPU and RAM of 16.0 GB was utilized.

STW was obtained from the forward portion of the timed-up-and-go (TUG) test which is a standard clinical assessment involving standing, walking 3 m, turning around, walking back, and sitting down [22]. During the trials subjects were initialized in a seated position on a backless, armless, height adjustable chair, with hands resting on their lap and the hip and knee flexed at approximately 90° (Fig. 1). This data collection

procedure was performed in the Mocap laboratory at Monash University Malaysia. Informed consent was obtained from all subjects; with ethical approval granted by the university's human research ethics committee. This STW Mocap dataset is presented in [22] and publicly accessible on the Bridges repository with the DOI: https://doi.org/10.26180/24515092.v4

B. Data Preprocessing

STW was identified from quiet-sitting till the first toe-off of the stance foot (Fig. 1) and was extracted from the TUG trials based on the STW transition phases described by Buckley et al. [23] and detailed in Perera et al. [4]. The raw Mocap and ground reaction force (GRF) data were processed using Qualisys Track Manager, the operational software for the Qualisys Mocap system. Following marker identification and gap filling, the data was exported in c3d format. Both Mocap and GRF data were filtered to eliminate noise, motion artifacts, and for smoothing, using a second-order Butterworth low pass filter at cutoff frequencies of 5 Hz and 10 Hz, respectively [24], [25]. These frequencies were determined based on the 99% occupied bandwidth from a Fast Fourier Transform.

Subsequently, the STW c3d files were imported into OpenSim 4.2 for biomechanical analysis utilizing the lower body Gait 2392 musculoskeletal model (Fig. 1). Scaling was performed to match subject anthropometry to the model on a subject specific basis, verified by ensuring the maximum marker and root mean square error (RMSE) were less than 1 cm and 2 cm respectively, as recommended by OpenSim [26]. Inverse kinematics was performed to derive the hip and knee joint angles followed by inverse dynamics for the modelled ground truth joint torques [27]. Further, a body kinematics analysis was run to derive HCOM velocity, while joint torque was normalized with respect to bodyweight and joint angles to 0° when upright.

C. Sit-to-Walk Strategy Identification

The hip and knee joint angles at gait initiation (GI) were used to classify the three STW strategies (forward continuation, balance and STSW). A K-means clustering algorithm was applied following the methodology detailed in our previous work [4] and is presented in Fig. 2, considering the 65 subject sample. To validate these results gap statistics and silhouette analysis were conducted, while each STW trial in the dataset was identified by its respective strategy. Gap statistics calculates the log of the pooled within-cluster sum of squared distances for each data point with respect to its centroid. It saturated at three clusters, thus verifying the classification into three generalized STW strategies [28]. Similarly, silhouette analysis indicates cluster cohesion and separation and resulted in all data points being greater than zero. This establishes that all STW trials were correctly assigned to their cluster [29].

D. Training Dataset Preparation

After preprocessing and strategy classification, the STW dataset was constructed using the derived biomechanical data (Fig. 3). This dataset comprised three input features - hip and knee joint angles with HCOM velocity; and two output features - modelled hip and knee torques (via inverse dynamics). Each STW trial contained a varying number of data points and was therefore time-normalized to a uniform 201 points, which can be represented as a percentage of the STW cycle. The number of points ensures a sampling frequency above 50 Hz is maintained as recommended for dynamic Mocap applications [30]. To prevent the networks from training on the same subjects and strategies consecutively the STW trials were shuffled between inter and intra subjects. Min-max feature scaling (normalization) was also applied to all variables, scaling them between zero and one. Thereafter, the dataset was split for training (70% with 230 STW trials), validation (20% with 65 STW trials), and unseen testing (10% with 30 STW trials). The test set included an equal number of trials for each of the three strategies enabling a fair evaluation of the LSTMs across all STW executions.

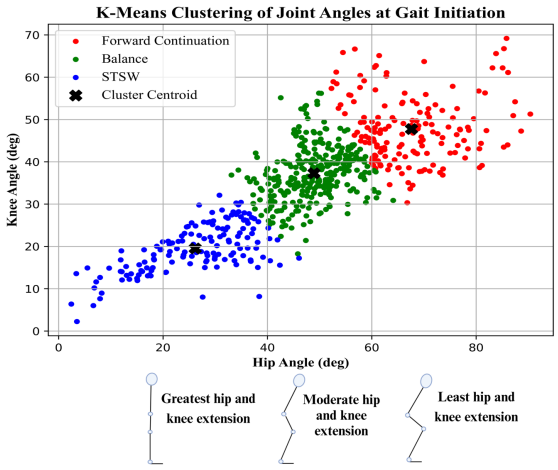

Fig. 2. K-means clustering for classification of the three sit-to-walk (STW) strategies. It was performed for all 65 subjects, where each data point represents the hip and knee joint angles at gait initiation (GI), as described in [4]. Forward continuation in red has the least hip and knee extension, followed by balance in green, and then sit-to-stand-wand-walk (STSW) in blue with the greatest hip and knee extension (almost upright at GI).

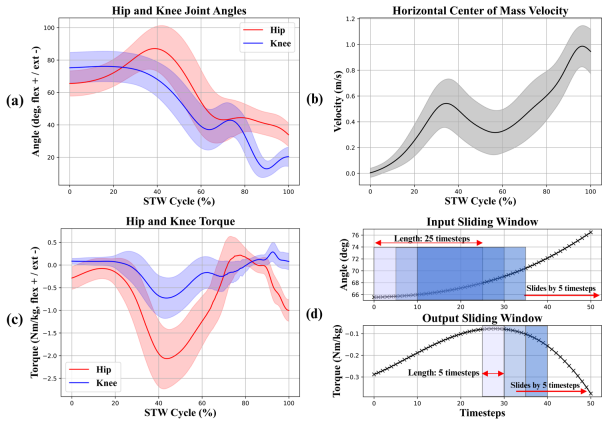

Fig. 3. The sit-to-walk (STW) dataset for model training consisting of (a) hip and knee joint angles, (b) horizontal center of mass velocity, and (c) modelled hip and knee joint torques. The solid lines represent the mean and the shaded regions represent their standard deviation. Additionally, (d) illustrates the input and output sliding windows.

Windowing was used to segment the data into a collection of input and output sliding windows. Model performance was observed for varying window lengths where the optimal values were determined as 25 input and five output timesteps (Fig. 3), as also recommended by Zaroug et al., [20]. This resulted in training and test datasets containing 25 samples with three features for the input and five samples with two features for the output. Windowing allows real-time torque prediction and mirrors the operation of a torque controller as the model predicts five output timesteps and then slides forward, thereby continuously forecasting future torque for the STW cycle. It enables the network to tailor the output torque based on the real-time variation in inputs, relative to the STW strategies and executions.

E. Long Short-Term Memory Architectures

STW data is timeseries in nature containing both temporal and spatial features, hence LSTM neural networks are well suited for STW torque prediction. This architecture was selected as it excels in multi-timestep sequence-to-sequence tasks, which can map input joint angles and HCOM velocity directly to predicted output hip and knee torques of different sequence lengths [31], [32]. LSTMs are also effective at analyzing temporal and sequential STW data, where data points are dependent and influenced by their historical values. To generate strategy specific torque trajectories and controllers, long and short-term temporal relationships in the data need to be captured, and LSTMs achieve this distinctive functionality by utilizing memory cells with forget, input, and output gates. Overall, these networks produce strong predictions of future torque by leveraging the current input with prior knowledge, compared to alternate architectures such as feedforward artificial neural networks or CNNs. This is supported by existing literature which benchmarked the LSTM against other neural models and showcased its effectiveness and applicability for predicting timeseries joint torques, kinematics, and kinetics [7], [15], [18], [20], [33].

Three LSTM networks (Fig. 4) were trained, tested, and compared for the multi-variate, multi-timestep forecasting of hip and knee torques. These include a (1) Vanilla LSTM, (2) Encoder-decoder LSTM and (3) CNN-LSTM, which are commonly used in sequence-to-sequence predictions [3], [7], [15], [20]. The vanilla LSTM serves as the base model with only LSTM and dense layers and was selected as a benchmark for evaluating the performance of the other models. The encoder-decoder was selected based on literature as it performs temporal feature extraction using the encoder layer. This network summarizes the input data before predicting an output, where the separation into an encoder and decoder allows complex long-term dependencies to be captured compared to the vanilla LSTM [15], [31], [32]. To further investigate spatial dependencies for torque prediction in the STW dataset, a CNN-LSTM was selected that extracts spatial features using convolution operators. This model captures both spatial and temporal relationships compared to the encoder-decoder LSTM [3], [33]. Comparing these models allows evaluation of the forecasting methods, for an accurate approach to generate strategy and user oriented joint torque trajectories. This

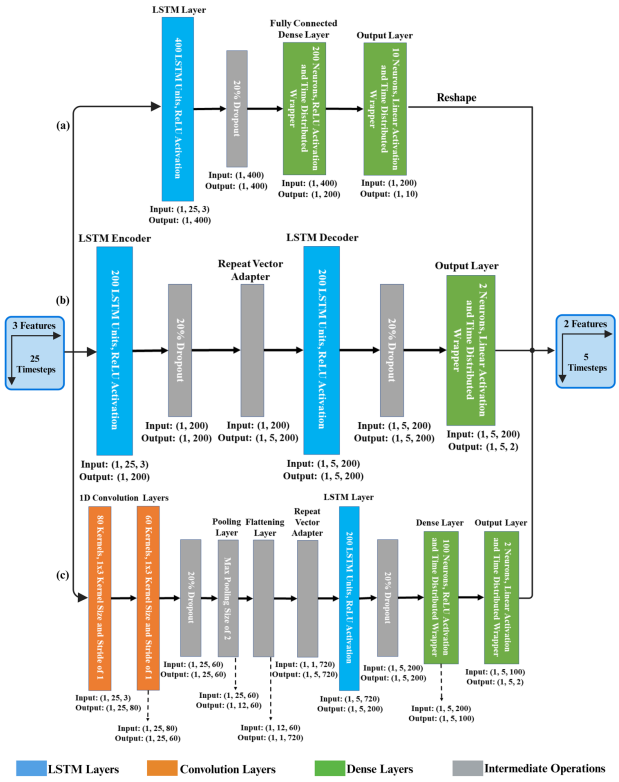

Fig. 4. The long short-term memory (LSTM) architectures consisting of a (a) vanilla LSTM (b) encoder-decoder LSTM and (c) convolution neural network (CNN)–LSTM. The data input/output dimensions for each layer are also presented.

also enables an assessment of whether the added network layers improve performance and the extent of model generalizability to unseen data.

The layers, neurons, and hyperparameters in all three LSTM architectures (Fig. 4) were selected empirically for optimal and stable performance. Model iterations were trained by systematically changing the hyperparameters until no further improvement in model performance was observed or overfitting took place. The tuned hyperparameters and intervals considered during training are presented in Table I. The vanilla LSTM used a single LSTM and fully connected dense layer with 400 and 200 neurons, respectively. The LSTM layer accepts the input joint angles and HCOM velocity, learns temporal relationships, and then forms an internal representation via a fixed length feature vector. The dense layer interprets this to provide an averaged output prediction, while also acting as a bottleneck that allows the model to learn a more compact representation of the data and prevent overfitting [34]. This model uses a vector output method which accepts multivariate inputs but only produces a single vector output. Hence, the predicted hip and knee torques are concatenated into a single vector, which is then reshaped accordingly. Alternatively, the encoder-decoder LSTM architecture consists of two LSTM layers, each with 200 neurons. The encoder is responsible for learning temporal relationships to form a fixed length feature vector. This passes through a repeat vector adapter which repeats the encoded sequence for each required timestep in the output sequence, to match the decoder's required format.

TABLE I
THE TUNED HYPERPARAMETERS, TRAINING INTERVALS, AND OPTIMAL VALUES FOR THE THREE LONG SHORT-TERM MEMORY (LSTM) ARCHITECTURE

Neural Network	Hyperparameters	Layers and Trained Intervals	Optimal Value
Vanilla	Weight initializer seed	[25, 300]	250
	LSTM layers and neurons	[25, 500] neurons with [1, 2] layers.	400 neurons with a single LSTM layer.
	Dense layers and neurons	[50, 400] neurons with [0, 2] layers.	200 neurons with a single dense layer.
Encoder-decoder	Weight initializer seed	[25, 300]	80
	LSTM layers and neurons	[25, 500] neurons with [1, 2] encoder-decoder layers.	200 neurons with a single encoder-decoder layer.
	Dense layers and neurons	[25, 200] neurons with [0, 1] layers.	None
CNN-LSTM	Weight initializer seed	[25, 300]	80
	1D convolution layers and kernels	[5, 100] kernels with [1, 3] convolution layers.	80 and 60 kernels for 2 convolution layers.
	Kernel size	1x3	1x3
	Stride and padding	[1, 3] stride length with same or valid padding.	Stride of 1 with same padding.
	Max pooling	[1, 4] pooling size with [1, 2] layers.	Pooling size of 2 with a single layer.
	LSTM layers and neurons	[25, 300] neurons with [1, 2] layers.	200 neurons with a single LSTM layer.
	Dense layers and neurons	[25, 200] neurons with [0, 1] layers.	100 neurons with a single dense layer.
Common parameters	Feature scaling (normalization)	MinMax[0, 1], MinMax[-1, 1] and StandardScaler	MinMax[0, 1]
	Activation function	Sigmoid, ReLU, Tanh and Leaky ReLU.	ReLU
	Loss function	MSE and MAE	MSE
	Learning rate	[$1x10^{-3}$, $1x10^{-5}$]	$1x10^{-3}$
	Batch size	[8, 256]	16
	Dropout	[20, 60]	20
	Sliding window timesteps	Input: [5, 40]	Input: 25
		Output: [2, 30]	Output: 5

The decoder LSTM unpacks this information to make a prediction for each specified output timestep. A dense output layer (two neurons) then generates the hip and knee torque trajectories utilizing a time distributed wrapper. This predicts each timestep of the output sequence independently, using the same set of weights to every temporal slice of the input sequence [35].

The CNN-LSTM combines convolution layers for spatial information with a LSTM network for temporal interpretation. A 1D convolution operator extracts spatial relationships along the time axis, which is projected onto feature maps based on stride length and kernel size [33], [36]. Two consecutive convolution layers were used with 80 and 60 filters and 'same' zero padding to retain input dimensions. A stride length of one and a 1×3 kernel size were selected to consider all three model inputs at every time-dependent data point.

Rectified linear units (ReLU) account for non-linear behavior in the model, while a max pooling layer (1×2 pooling kernel size) accounts for spatial variance, reduces dimensionality, and preserves the key features obtained from convolution [36], [37]. Next, a flattening layer takes each row of the output from the pooling layer to form a column vector, which then passes through a repeat vector adapter and into the LSTM layer. Temporal predictions are made for each specified output timestep and averaged through a fully connected dense layer with 100 neurons and a time distributed wrapper. The output layer then generates the final hip and knee torques during STW.

F. Model Optimization and Training

The three LSTMs were trained empirically, with each network undergoing 10 independent repetitions to account for their stochastic nature and maintain robustness [15]. The Adam optimizer was employed with a learning rate of 1×10^{-3}, coupled with mean squared error (MSE) as the loss function. To prevent overfitting a dropout rate of 20% was applied along with early stopping during training based on the minimum validation loss. The patience parameter was set to 20 with a maximum of 300 epochs. Further, a batch size of 16 was utilized with a He-Normal weight initializer, to maintain uniform training between model repetitions and reduce the impact of vanishing/exploding gradients. ReLU activation functions were applied to all layers, except for the output layer which had linear activation and two neurons for predicting hip and knee torques.

G. Model Evaluation

1) Walk Forward Testing: Walk forward testing was implemented to assess each network's performance on the unseen test data. This approach was selected due to the temporal

TABLE II
THE SIMPLE MAIN EFFECTS ANALYSIS AND THE CORRESPONDING TUKEY HONEST SIGNIFICANT DIFFERENCE POST-HOC TESTS

Model	Strategy	RMSE (Nm/kg)		Spearman's Correlation (%)	
		Hip	Knee	Hip	Knee
Simple Main Effects (Two-way ANOVA)					
	Forward	***P*<0.001**	**0.015**	***P*<0.001**	0.333
	Balance	0.083	0.228	0.85	0.745
	STSW	**0.038**	0.304	0.379	**0.038**
Post-hoc Tukey HSD (Bonferroni Correction)					
Vanila vs. Encoder	Forward	1	1	1	-
Vanila vs. CNN	Forward	*P*<0.001	0.094	*P*<0.001	-
Encoder vs. CNN	Forward	*P*<0.001	0.017	*P*<0.001	-
Vanila vs. Encoder	Balance	-	-	-	-
Vanila vs. CNN	Balance	-	-	-	-
Encoder vs. CNN	Balance	-	-	-	-
Vanila vs. Encoder	STSW	0.801	-	-	0.101
Vanila vs. CNN	STSW	0.423	-	-	1
Encoder vs. CNN	STSW	0.033	-	-	0.043

nature of the torque trajectories, where each point is not an independent observation [20]. During testing, the model forecasts future hip and knee torque based on a 'history' variable (prior knowledge) that includes inputs appended from the previous 25 timesteps, as determined to be the optimal input window. The model then walks forward by five timesteps (optimal output window) as new input readings become available, continuously predicting torque as expected for the operation of a real-time torque controller [20], [38]. Torque predictions are individually generated for each STW trial in the test set, identified by their labelled strategy. The outputs are inversely scaled to reverse the Min-Max feature scaling and obtain the final hip and knee flexion/extension torques, normalized against subject bodyweight. Performance metrics are then calculated and averaged for the overall model performance.

2) Performance Metrics: To evaluate the LSTMs, the predicted torque trajectories were compared against the modelled ground truth values (via inverse dynamics) using standard metrics. RMSE (1) measures the prediction accuracy, with smaller values indicating a closer match between predicted and modelled torques. Spearman's correlation (2) was also calculated to measure the strength of the monotonic relationship between the predicted and modelled trajectories, where a larger coefficient describes a closer match in trend [3].

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} (x_i - \hat{x}_i)^2}{n}} \tag{1}$$

$$\rho = 1 - \frac{6 \sum_{i=1}^{n} d_i^2}{n(n^2 - 1)} \tag{2}$$

Equation (1) describes the RMSE, and (2) represents the Spearman's rank correlation coefficient (ρ). The modelled and predicted joint torques are represented by x_i and $\hat{x}_i$, respectively while di is the difference between their ranks. Additionally, n represents the total number of data points in each STW cycle.

3) Statistical Analysis: Due to the varying strategy-wise torque requirements, the models must encompass all three strategies to be representative of STW executions. Thus, the performance of the LSTMs were evaluated against these STW strategies, through a statistical analysis using SPSS Statistics (IBM) with an α of 0.05. The independent variables were model type (vanilla, encoder-decoder, CNN-LSTM) and strategy (forward continuation, balance, STSW), while the dependent variables were the hip and knee torque performance metrics (RMSE and Spearman's correlation); with data from the 10 independently trained repetitions of each model.

Normality was tested using a Shapiro-Wilk test across both independent variables, resulting in a parametric distribution. A statistically significant interaction (P <0.05) between model type and strategy was observed, leading to a simple main effects analysis (two-way ANOVA design); to compare mean performance between the LSTMs for each individual strategy. The corresponding post-hoc test applied was Tukey Honest Significant Difference (HSD) with a Bonferroni correction to minimize Type I error. Additionally, to quantify model repeatability and reliability, the intraclass correlation coefficient (ICC) using a two-way mixed effects, single measurement, absolute agreement model was implemented for each performance metric [39].

III. RESULTS

From the developed LSTMs, the average training and validation losses across all models were 0.0421 ± 0.0040 and 0.0491 ± 0.0019, respectively. Similarly, the encoder-decoder and CNN-LSTM produced robust training and validation accuracies above 90%, whereas the vanilla LSTM exhibited lower accuracy above 60%. This drop in performance can be attributed to the vector output method, where both hip and knee output torques were concatenated into a single vector leading to a misleading measure of accuracy. Table II summarizes the results of the simple main effects analysis and the corresponding Tukey HSD post-hoc test. Likewise, Table III presents the average RMSE and Spearman's correlation coefficients for each LSTM, per strategy. Across the networks, performance was relatively strong with overall hip

TABLE III
MODEL PERFORMANCE METRICS AND INTRACLASS CORRELATION COEFFICIENTS

Model	Strategy	RMSE (Nm/kg)		Spearman's Correlation (%)	
		Hip	Knee	Hip	Knee
Model Performance (Mean ± Standard Deviation)					
Vanilla	Overall	0.25 ± 0.07	0.15 ± 0.02	93.42 ± 2.88	84.27 ± 3.63
	Forward	0.33 ± 0.03	0.17 ± 0.01	89.61 ± 1.17	87.54 ± 1.08
	Balance	0.17 ± 0.01	0.13 ± 0.004	95.75 ± 0.71	85.30 ± 0.92
	STSW	0.25 ± 0.02	0.16 ± 0.01	94.89 ± 0.54	79.98 ± 2.64
Encoder-decoder	Overall	0.24 ± 0.07	0.15 ± 0.02	93.43 ± 2.86	84.83 ± 2.96
	Forward	0.32 ± 0.01	0.16 ± 0.01	89.58 ± 0.99	87.68 ± 0.51
	Balance	0.16 ± 0.01	0.13 ± 0.003	95.64 ± 0.50	85.06 ± 0.81
	STSW	0.24 ± 0.01	0.16 ± 0.01	95.08 ± 0.45	81.75 ± 2.78
CNN-LSTM	Overall	0.28 ± 0.09	0.16 ± 0.02	92.14 ± 4.30	83.69 ± 3.50
	Forward	0.38 ± 0.04	0.18 ± 0.02	86.30 ± 1.18	86.56 ± 0.91
	Balance	0.18 ± 0.01	0.14 ± 0.003	95.54 ± 0.67	84.68 ± 1.04
	STSW	0.26 ± 0.02	0.16 ± 0.01	94.58 ± 0.64	79.82 ± 3.27
Intraclass Correlation Coefficient (ICC)					
Vanilla		0.94	0.76	0.94	0.83
Encoder-Decoder		0.98	0.82	0.96	0.75
CNN-LSTM		0.94	0.68	0.97	0.74

and knee RMSE being less than 0.28 Nm/kg and 0.16 Nm/kg, respectively, and the overall correlations exceeding 92.14% and 83.69%, respectively.

From Table II, statistically significant differences ($P < 0.05$) in model performance were observed amongst forward continuation and STSW strategies, but none across the balance strategy. This indicates that all models predicted balance torques with similar error and correlations to the ground truth. The post-hoc tests revealed statistically significant differences in model performance, primarily between the encoder-decoder and the other models. Considering forward continuation, the encoder-decoder exhibited the lowest hip and knee RMSE with 0.32 ± 0.01 Nm/kg and 0.16 ± 0.01 Nm/kg, respectively; followed by the vanilla and then CNN-LSTM. The encoder-decoder also displayed the highest knee correlation with 87.68 ± 0.51%, while the vanilla LSTM had a marginally higher hip correlation of 89.61 ± 1.17%, compared to the encoder-decoder's 89.58 ± 0.99%. For STSW, the encoder-decoder had the lowest hip RMSE of 0.24 ± 0.01 Nm/kg but equivalent knee RMSE (0.16 ± 0.01 Nm/kg) with the other models. Further, it also exhibited the highest statistically significant knee correlation of 81.75 ± 2.78%, with no significant differences in hip correlation (95.08 ± 0.45%). These results are reflected in Fig. 5, which shows the predicted and modelled hip and knee torques for the three trained LSTM networks per strategy.

Across all LSTMs and strategies, the hip displayed greater RMSE and Spearman's correlation showing a closer match in trend but higher point-to-point predictive error, compared to knee torque prediction. Moreover, Table III presents the ICCs for each model with all values being greater than or equal to 0.75 indicating good strength; except for knee RMSE of the CNN-LSTM (0.68) which showed moderate strength [39]. Overall, these ICCs reflect a good degree of correlation within the 10 trained LSTM repetitions, thus showcasing good model reliability and repeatability.

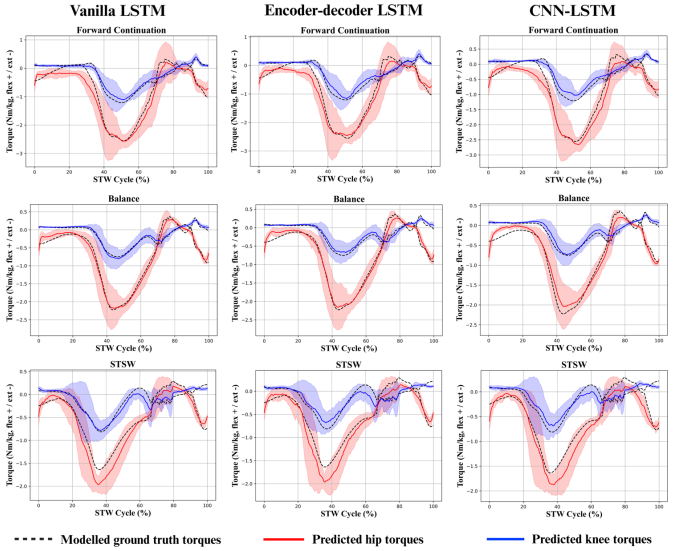

Fig. 5. The modelled (black dashed) and predicted hip (red) and knee (blue) torques from the three trained long short-term memory (LSTM) models per sit-to-walk (STW) strategy. The solid lines represent the mean torques across the test dataset and the shaded areas are their standard deviations.

The effect of varying the sliding window size was explored, as presented in Table I. For the input, windows below 10 timesteps produced a drop in performance while windows above 35 timesteps led to overfitting. In the output window there was a drop in performance for windows above five timesteps. Overall, reducing the input window and increasing the output window led to a reduction in model performance and overfitting. Larger input windows capture more temporal information but result in slower prediction times, while larger output windows predict more timesteps at the cost of performance. Hence, the input and output windows were selected as 25 and five samples respectively, as also recommended to be the optimal prediction horizon to consistently forecast

movement trajectories [20]. Considering a minimum sampling frequency of 50 Hz for Mocap applications [30], [40] the output window can continuously forecast future trajectories every 100 ms, requiring an initial input window of 500 ms. These timesteps are suitable for real-time assistance as they sufficiently predict future torque [38], [40] but would change in timing based on the sampling frequency of the controllers used in assistive devices.

IV. Discussions

Three LSTMs were developed and evaluated for predicting hip and knee joint torques during STW, oriented towards execution strategies and subject anthropometry via normalization through bodyweight. An individual can perform any STW strategy at an instance, each with specific generalized biomechanics and torque requirements [4]. Therefore, the predicted torque should match each strategy to encompass all STW executions; and this is verified through evaluation against these strategies. Utilizing neural networks as torque controllers allows for a direct mapping of inputs (hip and knee joint angles with HCOM velocity) to the output torques without the need for intermediary processing or biomechanical models. This confers an advantage over traditional torque prediction methods such as ID or NMS modelling [5], [7]. Neural models can also provide strategy-specific and user-oriented assistance, for more effective application in rehabilitation or when aiding movement impairments [3], [7], [18]. However, drawbacks include the requirement for big data during training, with a variety of subject characteristics and demographics inclusive of a population. These models also have a black-box nature making it challenging to analyze input/output relationships [13], [41].

Regarding performance, hip torque prediction produced higher Spearman's correlation and RMSE (greater than 86% and less than 0.38 Nm/kg, respectively) compared to the knee (greater than 79% and less than 0.18 Nm/kg, respectively). This infers that the networks had a stronger grasp of capturing the trends in hip torque trajectories but with a higher point-to-point error. The LSTMs showed no statistically significant differences for the balance strategy, which had the lowest RMSE across all models. Forward continuation and STSW had statistically significant differences across the models with the encoder-decoder showcasing the lowest hip and knee RMSE (0.32 and 0.16 Nm/kg for forward continuation and 0.24 and 0.16 Nm/kg for STSW respectively). For the hyperparameters, the vanilla LSTM required 400 and 200 neurons in the LSTM and dense layers respectively (728610 trainable parameters), while the CNN-LSTM had two convolution layers (80 and 60 filters) with a single LSTM (200 neurons) and dense (100 neurons) layer (772362 trainable parameters). In contrast, the encoder-decoder only contained two LSTM layers with 200 neurons resulting in less computational requirements (484402 trainable parameters). The CNN-LSTM used convolution operators for spatial feature extraction yet exhibited weaker performance than the encoder-decoder which considered temporal features. This could highlight a lower dependence on spatial features with greater reliance on temporal relationships for torque prediction using kinematics. Thus, the encoder-decoder LSTM is the optimal model for STW torque prediction, using joint angles and HCOM velocity as inputs.

TABLE IV
Encoder-decoder Long Short-Term Memory Performance Per Age Group

Performance Metrics (Mean ± Standard Deviation)				
Age Group	RMSE (Nm/kg)		Spearman's Correlation (%)	
	Hip	Knee	Hip	Knee
Young	0.20 ± 0.04	0.13 ± 0.06	93.36 ± 3.52	84.20 ± 3.74
Middle-aged	0.39 ± 0.05	0.19 ± 0.01	86.15 ± 3.40	86.40 ± 2.29
Older	0.24 ± 0.07	0.18 ± 0.03	96.29 ± 1.85	78.00 ± 7.11

The predicted torques during STW serve as a baseline in producing the lift assistance a patient requires. Healthy adult torque profiles are instrumental in the design and evaluation of torque controllers, as they set the reference trajectory to be followed during motion [1], [11]. The input parameters (joint angles and HCOM velocity) used to generate the torque profiles in this study are distinguishing features of the developed networks. Such inputs are easily measured using wearable sensors such as IMUs or encoders and readily integrated with wearable assistive devices [8], [42]. This contrasts with alternate inputs from literature such as GRFs which require bulky and expensive Mocap systems [43], or SEMG signals which face issues due to noise, sensor placement, electrode-skin displacement, and sweat; resulting in difficulties for prolonged use [3], [44].

The training and test datasets contained a substantial sample size of 65 subjects spanning a wide age range from 19 to 73 years, and a near-equal male to female distribution. This accounts for variability in human movement, subject anthropometry, age, biological sex, and STW strategies. Therefore, considering the central limit theorem [45] the trained LSTMs are generalizable to a population, compared to other torque prediction models from literature which utilized smaller sample sizes ($n < 25$) with focused subject and age groups [3], [5], [7], [8], [15], [42], [43]. The encoder-decoder LSTM also produced strong performance per age group (Table IV), where young adults had lower RMSE (less than 0.20 Nm/kg), higher knee correlation but lower hip correlation in comparison to older adults. Forward continuation required larger peak torques with higher error in model performance compared to the other strategies. This variation in performance (including middle-aged adults) can be attributed to the subject distribution by age in the data. Balance and forward continuation were most observed in young adults as good movement confidence and postural control is required. Alternatively, middle-aged and older adults favor balance and STSW, which acts as a compensatory strategy to maintain stability during this dynamic motion. Therefore, balance is the most commonly utilized strategy amongst the age groups, where future work could include equalizing sample sizes across strategy and age group through data augmentation [4], [23]. Additionally, the STW trials were shuffled to minimize the influence of consecutive

exposure to the same subject. This ensures that the models do not only learn characteristics specific to an individual or their movement style.

Performance metrics were calculated against modelled torque obtained through gold standard inverse dynamics. Comparisons can also be drawn with existing torque prediction models. Zhang et al., [15] obtained an RMSE less than 0.24 for the lower limbs, during multiple motions including sit-to-stand with a LSTM using joint angles and SEMG inputs. Wang et al., [19] produced a normalized RMSE less than 15% and correlations greater than 85% for lower body joints during gait, using a LSTM with SEMG inputs. Likewise, Moreira et al., [3] obtained a 89% Spearman's correlation for ankle torque prediction with joint kinematic inputs and a CNN model. Alternatively, Schulte et al., [46] utilized a NMS model to predict knee torque during gait with a NRMSE of 14.3%, while Liu et al. [1] employed a mathematical model with a maximal knee torque error of 3.27 Nm. For the trained encoder-decoder, overall RMSE and correlations for the hip and knee were 0.24 and 0.15 Nm/kg, and 93.43 and 84.83%, respectively. The developed network aligns with literature demonstrating efficacy and comparability with existing state-of-the-art torque prediction models.

In future work, this LSTM can be utilized within the control architecture of assistive devices. It acts as a torque controller that produces reference torque trajectories oriented towards the STW strategies for a control loop (such as proportional-integral-derivative) and actuator. These personalized assistive devices would provide strategy specific torque, enabling a more natural application of assistance with improved safety and efficiency during rehabilitation or when aiding movement impairment [3], [19]. Generating benchmark torque profiles of healthy adults can help analyze and correct impaired motion, which is crucial for patients recovering from neuro-musculoskeletal impairment such as spinal cord injury or the effects of a stroke. Furthermore, these neural models could be expanded to generate torque trajectories for other joints and ADLs such as stair ascent/decent or gait; utilizing methods like clustering and transfer learning [15].

Even with the current model's performance certain limitations still need to be addressed. From Fig. 5, the peak torques during STSW were underpredicted for the knee and overestimated for the hip. For this, attention layers could be implemented to focus on important input timesteps such as those around peak values [47]. Additionally, weighted loss functions could be explored to assign larger weights for errors related to peak values compared to the other sequences. Joint torques were also predicted considering symmetry between the swing and stance legs. However, they would have slightly varying torque trajectories especially after GI and should be considered based on the lead and trailing foot during STW, when designing appropriate controllers. The LSTMs require a small burn-in time of 25 input timesteps which is utilized in memory to make the first prediction [7]. This should be considered when designing controllers as it impacts the real-time operation of assistive devices. A memory component could be used to retain context (stateful) when beginning torque prediction and minimize burn-in time.

V. Conclusion

In this study three LSTM models were investigated for joint torque prediction during STW, oriented towards movement strategies and subject anthropometry. To encompass all STW executions, the models were evaluated against these strategies with the encoder-decoder LSTM being the optimal. This is considering joint angles and HCOM velocity as inputs, which can be measured and integrated with wearable technology. The encoder-decoder showcased strong performance with low RMSE and computational resources, high correlations, good repeatability, and is generalizable to a population. A sliding window was utilized for real-time torque assistance with 25 input and five output timesteps. This allows the model to dynamically adapt to changes in STW executions and produces a smoother reaction to varying STW strategies [20], [38]. These torque predictions serve as reference trajectories in the control architecture of assistive devices. They determine the level of assistance a patient requires, are user and strategy specific, and result in personalized application for rehabilitation or treatments.

Acknowledgment

The authors thank Dr. Wen Shan Tan (Monash University) for his assistance in model training. Figures were created using BioRender.com. The sit-to-walk motion capture data used in this study is publicly available on the Bridges repository by Monash University, with the identifier: https://doi.org/10.26180/24515092.v4.

References

[1] X. Liu, Z. Zhou, J. Mai, and Q. Wang, "Real-time mode recognition based assistive torque control of bionic knee exoskeleton for sit-to-stand and stand-to-sit transitions," *Robot. Auto. Syst.*, vol. 119, pp. 209–220, Sep. 2019, doi: 10.1016/j.robot.2019.06.008.

[2] M. Xiloyannis et al., "Soft robotic suits: State of the art, core technologies, and open challenges," *IEEE Trans. Robot.*, vol. 38, no. 3, pp. 1343–1362, Jun. 2022, doi: 10.1109/TRO.2021.3084466.

[3] L. Moreira, J. Figueiredo, J. P. Vilas-Boas, and C. P. Santos, "Kinematics, speed, and anthropometry-based ankle joint torque estimation: A deep learning regression approach," *Machines*, vol. 9, no. 8, p. 154, Aug. 2021, doi: 10.3390/machines9080154.

[4] C. K. Perera, A. A. Gopalai, D. Gouwanda, S. A. Ahmad, and M. S. B. Salim, "Sit-to-walk strategy classification in healthy adults using hip and knee joint angles at gait initiation," *Sci. Rep.*, vol. 13, no. 1, Oct. 2023, Art. no. 1, doi: 10.1038/s41598-023-43148-0.

[5] L. Zhang, D. Soselia, R. Wang, and E. M. Gutierrez-Farewik, "Estimation of joint torque by EMG-driven neuromusculoskeletal models and LSTM networks," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 31, pp. 3722–3731, 2023, doi: 10.1109/TNSRE.2023.3315373.

[6] L. N. Awad, A. Esquenazi, G. E. Francisco, K. J. Nolan, and A. Jayaraman, "The ReWalk ReStore™ soft robotic exosuit: A multi-site clinical trial of the safety, reliability, and feasibility of exosuit-augmented post-stroke gait rehabilitation," *J. NeuroEng. Rehabil.*, vol. 17, no. 1, p. 80, Jun. 2020, doi: 10.1186/s12984-020-00702-5.

[7] H. C. Siu, J. Sloboda, R. J. McKindles, and L. A. Stirling, "A neural network estimation of ankle torques from electromyography and accelerometry," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 29, pp. 1624–1633, 2021, doi: 10.1109/TNSRE.2021.3104761.

[8] H. Dinovitzer, M. Shushtari, and A. Arami, "Accurate real-time joint torque estimation for dynamic prediction of human locomotion," *IEEE Trans. Biomed. Eng.*, vol. 70, no. 8, pp. 2289–2297, Aug. 2023, doi: 10.1109/TBME.2023.3240879.

[9] M. E. Roebroeck, C. A. M. Doorenbosch, J. Harlaar, R. Jacobs, and G. J. Lankhorst, "Biomechanics and muscular activity during sit-to-stand transfer," *Clin. Biomech.*, vol. 9, no. 4, pp. 235–244, Jul. 1994, doi: 10.1016/0268-0033(94)90004-3.

[10] A. Tsukahara, R. Kawanishi, Y. Hasegawa, and Y. Sankai, "Sit-to-stand and stand-to-sit transfer support for complete paraplegic patients with robot suit HAL," *Adv. Robot.*, vol. 24, no. 11, pp. 1615–1638, Jan. 2010, doi: 10.1163/016918610x512622.

[11] K. Schmidt et al., "The myosuit: Bi-articular anti-gravity exosuit that reduces hip extensor activity in sitting transfers," *Frontiers Neurorobot.*, vol. 11, Oct. 2017, Art. no. 57, doi: 10.3389/fnbot.2017.00057.

[12] C. M. Lara-Barrios, A. Blanco-Ortega, A. Abúndez-Pliego, and J. Colín-Ocampo, "Sit-to-stand simulation for torque estimation on lower limb joints," in *Proc. Int. Conf. Mechatronics, Electron. Automot. Eng. (ICMEAE)*, Nov. 2015, pp. 116–121, doi: 10.1109/ICMEAE.2015.33.

[13] L. Wang and T. S. Buchanan, "Prediction of joint moments using a neural network model of muscle activations from EMG signals," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 10, no. 1, pp. 30–37, Mar. 2002, doi: 10.1109/TNSRE.2002.1021584.

[14] 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, Nov. 2007, doi: 10.1109/TBME.2007.901024.

[15] L. Zhang, D. Soselia, R. Wang, and E. M. Gutierrez-Farewik, "Lower-limb joint torque prediction using LSTM neural networks and transfer learning," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 600–609, 2022, doi: 10.1109/TNSRE.2022.3156786.

[16] A. Kerr, B. Durward, and K. M. Kerr, "Defining phases for the sit-to-walk movement," *Clin. Biomech.*, vol. 19, no. 4, pp. 385–390, May 2004, doi: 10.1016/j.clinbiomech.2003.12.012.

[17] P. Dehail et al., "Kinematic and electromyographic analysis of rising from a chair during a 'sit-to-walk' task in elderly subjects: Role of strength," *Clin. Biomech.*, vol. 22, no. 10, pp. 1096–1103, Dec. 2007, doi: 10.1016/j.clinbiomech.2007.07.015.

[18] M. T. N. Truong, A. E. A. Ali, D. Owaki, and M. Hayashibe, "EMG-based estimation of lower limb joint angles and moments using long short-term memory network," *Sensors*, vol. 23, no. 6, p. 3331, Mar. 2023, doi: 10.3390/s23063331.

[19] M. Wang et al., "Lower limb joint torque prediction using long short-term memory network and Gaussian process regression," *Sensors*, vol. 23, no. 23, p. 9576, Dec. 2023, doi: 10.3390/s23239576.

[20] A. Zaroug, D. T. H. Lai, K. Mudie, and R. Begg, "Lower limb kinematics trajectory prediction using long short-term memory neural networks," *Frontiers Bioeng. Biotechnol.*, vol. 8, p. 362, May 2020, doi: 10.3389/fbioe.2020.00362.

[21] N. M. Petry, "A comparison of young, middle-aged, and older adult treatment-seeking pathological gamblers," *Gerontologist*, vol. 42, no. 1, pp. 92–99, Feb. 2002, doi: 10.1093/geront/42.1.92.

[22] C. K. Perera, Z. Hussain, M. Khant, A. A. Gopalai, D. Gouwanda, and S. A. Ahmad, "A motion capture dataset on human sitting to walking transitions," *Sci. Data*, vol. 11, no. 1, p. 878, Aug. 2024, doi: 10.1038/s41597-024-03740-z.

[23] T. Buckley, C. Pitsikoulis, E. Barthelemy, and C. J. Hass, "Age impairs sit-to-walk motor performance," *J. Biomech.*, vol. 42, no. 14, pp. 2318–2322, Oct. 2009, doi: 10.1016/j.jbiomech.2009.06.023.

[24] F. Crenna, G. B. Rossi, and M. Berardengo, "Filtering biomechanical signals in movement analysis," *Sensors*, vol. 21, no. 13, p. 4580, Jul. 2021, doi: 10.3390/s21134580.

[25] D. Winter, *Biomechanics and Motor Control of Human Movement*, 4th ed. Hoboken, NJ, USA: Wiley, Sep. 2009, doi: 10.1002/9780470549148.ch5.

[26] OpenSim. *Scaling Best Practices—OpenSim Documentation*. Accessed: Jul. 20, 2021. [Online]. Available: https://simtk-confluence.stanford.edu/display/OpenSim/_Scaling+Best+Practices

[27] OpenSim. *User's Guide—OpenSim Documentation*. Accessed: Jul. 28, 2021. [Online]. Available: https://simtk-confluence.stanford.edu/display/OpenSim/User%27s+Guide

[28] R. Tibshirani, G. Walther, and T. Hastie, "Estimating the number of clusters in a data set via the gap statistic," *J. Roy. Stat. Soc. B, Stat. Methodol.*, vol. 63, no. 2, pp. 411–423, May 2001.

[29] M. Shutaywi and N. N. Kachouie, "Silhouette analysis for performance evaluation in machine learning with applications to clustering," *Entropy*, vol. 23, no. 6, p. 759, Jun. 2021, doi: 10.3390/e23060759.

[30] E. van der Kruk and M. M. Reijne, "Accuracy of human motion capture systems for sport applications; state-of-the-art review," *Eur. J. Sport Sci.*, vol. 18, no. 6, pp. 806–819, Jul. 2018, doi: 10.1080/17461391.2018.1463397.

[31] K. Cho et al., "Learning phrase representations using RNN encoder–decoder for statistical machine translation," Sep. 2014, *arXiv:1406.1078*.

[32] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," Dec. 2014, *arXiv:1409.3215*.

[33] K. Xia, J. Huang, and H. Wang, "LSTM-CNN architecture for human activity recognition," *IEEE Access*, vol. 8, pp. 56855–56866, 2020, doi: 10.1109/ACCESS.2020.2982225.

[34] K. Greff, R. K. Srivastava, J. Koutník, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A search space Odyssey," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2222–2232, Oct. 2017, doi: 10.1109/TNNLS.2016.2582924.

[35] T. Hori, W. Wang, Y. Koji, C. Hori, B. Harsham, and J. R. Hershey, "Adversarial training and decoding strategies for end-to-end neural conversation models," *Comput. Speech Lang.*, vol. 54, pp. 122–139, Mar. 2019, doi: 10.1016/j.csl.2018.08.006.

[36] R. Rick and L. Berton, "Energy forecasting model based on CNN-LSTM-AE for many time series with unequal lengths," *Eng. Appl. Artif. Intell.*, vol. 113, Aug. 2022, Art. no. 104998, doi: 10.1016/j.engappai.2022.104998.

[37] T. Y. Kim and S. B. Cho, "Predicting residential energy consumption using CNN-LSTM neural networks," *Energy*, vol. 182, pp. 72–81, Sep. 2019, doi: 10.1016/j.energy.2019.05.230.

[38] O. Banos, J.-M. Galvez, M. Damas, H. Pomares, and I. Rojas, "Window size impact in human activity recognition," *Sensors*, vol. 14, no. 4, pp. 6474–6499, Apr. 2014, doi: 10.3390/s140406474.

[39] T. K. Koo and M. Y. Li, "A guideline of selecting and reporting intraclass correlation coefficients for reliability research," *J. Chiropractic Med.*, vol. 15, no. 2, pp. 155–163, Jun. 2016, doi: 10.1016/j.jcm.2016.02.012.

[40] G. Wang, Q. Li, L. Wang, W. Wang, M. Wu, and T. Liu, "Impact of sliding window length in indoor human motion modes and pose pattern recognition based on smartphone sensors," *Sensors*, vol. 18, no. 6, p. 1965, Jun. 2018, doi: 10.3390/s18061965.

[41] Y. Li, W. Chen, H. Yang, J. Li, and N. Zheng, "Joint torque closed-loop estimation using NARX neural network based on sEMG signals," *IEEE Access*, vol. 8, pp. 213636–213646, 2020, doi: 10.1109/ACCESS.2020.3039983.

[42] Z. Altai, I. Boukhennoufa, X. Zhai, A. Phillips, J. Moran, and B. X. W. Liew, "Performance of multiple neural networks in predicting lower limb joint moments using wearable sensors," *Frontiers Bioeng. Biotechnol.*, vol. 11, Jul. 2023, Art. no. 1215770. Accessed: Jan. 5, 2024. [Online]. Available: https://www.frontiersin.org/articles/10.3389/fbioe.2023.1215770

[43] M. V. McCabe, D. W. Van Citters, and R. M. Chapman, "Hip joint angles and moments during stair ascent using neural networks and wearable sensors," *Bioengineering*, vol. 10, no. 7, p. 784, Jun. 2023, doi: 10.3390/bioengineering10070784.

[44] C. Su, S. Chen, H. Jiang, and Y. Chen, "Ankle joint torque prediction based on surface electromyographic and angular velocity signals," *IEEE Access*, vol. 8, pp. 217681–217687, 2020, doi: 10.1109/ACCESS.2020.3040820.

[45] S. G. Kwak and J. H. Kim, "Central limit theorem: The cornerstone of modern statistics," *Korean J. Anesthesiol.*, vol. 70, no. 2, pp. 144–156, Apr. 2017, doi: 10.4097/kjae.2017.70.2.144.

[46] R. V. Schulte, M. Zondag, J. H. Buurke, and E. C. Prinsen, "Multi-day EMG-based knee joint torque estimation using hybrid neuromusculoskeletal modelling and convolutional neural networks," *Frontiers Robot. AI*, vol. 9, Apr. 2022, Art. no. 869476. Accessed: Dec. 30, 2023. [Online]. Available: https://www.frontiersin.org/articles/10.3389/frobt.2022.869476

[47] V. Nguyen and O. Bodenreider, "Adding an attention layer improves the performance of a neural network architecture for synonymy prediction in the UMLS metathesaurus," *Stud. Health Technol. Inf.*, vol. 290, pp. 116–119, Jun. 2022, doi: 10.3233/SHTI220043.