2020 IEEE International Conference on
Autonomous Robot Systems and Competitions (ICARSC)
April 15-16, Ponta Delgada, Portugal

AI-based Reference Ankle Joint Torque Trajectory Generation for Robotic Gait Assistance: First Steps

Luís Moreira
University of Minho
Guimarães, Portugal
luis.lihts@hotmail.com

Sara M. Cerqueira, Joana Figueiredo
University of Minho
Guimarães, Portugal
saracerqueira1996@gmail.com
id6003@alunos.uminho.pt

João Vilas-Boas
University of Porto
Porto, Portugal
jpvb@fade.up.pt

Cristina P. Santos
University of Minho
Guimarães, Portugal
cristina@dei.uminho.pt

***Abstract*— Robotic-based gait rehabilitation and assistance have been growing to augment and to recover motor function in subjects with lower limb impairments. There is interest in developing user-oriented control strategies to provide personalized assistance. However, it is still needed to set the healthy user-oriented reference joint trajectories, namely, reference ankle joint torque, that would be desired under healthy conditions. Considering the potential of Artificial Intelligence (AI) algorithms to model nonlinear relationships of the walking motion, this study implements and compares two offline AI-based regression models (Multilayer Perceptron and Long-Short Term Memory-LSTM) to generate healthy reference ankle joint torques oriented to subjects with a body height ranging from 1.51 to 1.83 m, body mass from 52.0 to 83.7 kg and walking in a flat surface with a walking speed from 1.0 to 4.0 km/h. The best results were achieved for the LSTM, reaching a Goodness of Fit and a Normalized Root Mean Square Error of 79.6 % and 4.31 %, respectively. The findings showed that the implemented LSTM has the potential to be integrated into control architectures of robotic assistive devices to accurately estimate healthy user-oriented reference ankle joint torque trajectories, which are needed in personalized and Assist-As-Needed conditions. Future challenges involve the exploration of other regression models and the reference torque prediction for remaining lower limb joints, considering a wider range of body masses, heights, walking speeds, and locomotion modes.**

Keywords—Ankle Joint Torque Prediction, Artificial Intelligence, Control Strategies, Regression Models, Robotic Gait Rehabilitation

I. INTRODUCTION

Lower limb disabilities correspond to one of the major consequences of neurological diseases. Regarding stroke events, according to [1], around 63 % of the stroke survivors cannot walk without external support and consequently, their daily life activities are severely compromised [2].

In the last decade, the number of studies focused on robotic-based gait rehabilitation and assistance managed by assistive control strategies emerged, producing different robot-assisted training modes destined to the recovery stage of each patient [3].

This work has been supported by the FEDER Funds through the Programa Operacional Regional do Norte and national funds from Fundação para a Ciência e Tecnologia with the project SmartOs under Grant NORTE-01-0145-FEDER-030386, and through the COMPETE 2020—Programa Operacional Competitividade e Internacionalização (POCI)—with the Reference Project under Grant POCI-01-0145-FEDER-006941.

In the last decade, the number of studies focused on robotic-based gait rehabilitation and assistance managed by assistive control strategies emerged, producing different robot-assisted training modes destined to the recovery stage of each patient [3]. With these methods, physicians can be released from the heavy training therapies and the patient's recovery status can be objectively analyzed through the data recorded during the training session.

According to [4], [5], the patient should receive: ***(i)*** passive locomotion modes based on predefined joint trajectories in early rehabilitation stages, to reduce muscle atrophy and augment the movement capacity; ***(ii)*** active locomotion modes, encouraging the patient's participation when certain levels of strength are achieved. Regarding the passive locomotion modes, typically managed by position assistive control strategies, predefined reference joint position trajectories are commonly adopted from datasets available on the Literature ([6]–[8]) or they are determined by user-oriented models [9], [10]. The active locomotion modes have been the target of many studies well-reviewed in [3], towards Assist-As-Needed (AAN) control strategies, comprehending Force, Impedance, EMG-based control strategies. Recently, there is a considerable motivation in the development of adaptive and user-oriented AAN EMG-based control strategies ([11]–[14]), stimulated by their capacity to: ***(i)*** invoke the patient's participation; ***(ii)*** detect the user's motion intention based on EMG signals; and ***(iii)*** offer an amount of assistance as much as needed, i.e., providing AAN training.

Nonetheless, in AAN control strategies, to determine the adequate assistance level to provide to each patient, it is required to firstly set a user-oriented reference joint trajectory that would be desired under healthy conditions, considering the patient's anthropometry. This estimated reference trajectory should be correctly generated since it represents the desired trajectory that each patient should achieve over therapy sessions. On the other hand, the development of control strategies based on EMG signals typically requires the use of torque controllers [15]–[17]. Thus, to combine EMG-based and AAN control strategies, it is necessary to generate a healthy user-oriented reference joint trajectory, namely, reference joint torque, based on well-known data from each subject. With this, personalized robotic gait support can be provided, making the assistance process more adequate for each subject.

To the best knowledge of the authors, there is no evidence of studies able to estimate healthy reference joint torque

Authorized licensed use limited to: Universita Studi di Torino - Dipartimento Di Informatica. Downloaded on September 12,2020 at 09:57:59 UTC from IEEE Xplore. Restrictions apply.

trajectories during the entire gait cycle based on well-known data, such as walking speed, kinematic and anthropometric data, avoiding prior data collections. Note that kinematic data do not require prior data collections since they can be generated using algorithms developed in [9], [10].

The goal of the present study is the offline estimation of healthy user-oriented reference ankle joint torques, based on ankle joint kinematics (on the sagittal plane), walking speed and anthropometric data (body height and mass) from each user. The ankle joint was chosen, since this is the joint that requires more assistance when compared to the remaining lower limb joints (knee and hip joint), commonly dealing with drop foot conditions as a consequence of neurological diseases [18]. To realize that, we implemented, validated and compared two regression models based on Artificial Intelligence (AI) algorithms, namely, a Multilayer Perceptron (MLP) and a Long-Short Term Memory (LSTM) neural networks since they have potential to model nonlinear relationships of data from the walking motion [19]–[21]. The proper determination of reference ankle joint torques is of utmost importance to provide the most suitable assistance level oriented the needs of each user, aiming a rehabilitation process more effective.

The paper is organized as follows. Section II presents a brief description of the used regression models and section III presents all the procedures to collect walking motion data, in order to validate the proposed models. Section IV exhibits the organization of the collected data for the regression models, as well as their implementation. Section V pertains to the results and the discussion of this study. Finally, section VI summarizes the main conclusions of the developed work, along with future perspectives.

II. Background of Used Artificial Intelligence Algorithms

Among a huge variability of AI algorithms already developed, MLP and LSTM neural networks have been widely used to solve regression problems [22]–[24]. In this study, these two models were implemented, trained and optimized in *MATLAB R2018b*.

A. Concepts of MLPNeural Network

MLP was chosen due to its reputable performance exhibited in a considerable quantity of studies already presented in the literature [23], [25]. This neural network is composed of **(*i*)** input layers containing independent variables; **(*ii*)** hidden layers incorporating activation functions; and **(*iii*)** output layers including dependent variables. Since all nodes of each layer are fully connected to the nodes of the next or previous layer, the MLP is called a feedforward neural network.

Backpropagated algorithms have been generally used as the training process of MLPs since the convergence of the neural network is improved [26]. The principle behind MLPs with backpropagation algorithms is based on the error gradient computation with respect to the weights. This error is backpropagated through the neural network based on the gradient descent techniques. Thus, the weights are updated according to the error between the predicted output and the real signal. This can enable the achievement of predictive values closer to the target [24].

Nonetheless, the gradient descent algorithms used to backpropagate the error can stick in a local minimum. To solve this problem, two new parameters can be considered, namely, learning rate and momentum. With these parameters, the effect of the error gradient on the weights update is controlled and it is possible to avoid local minimums [27]. However, the convergence and generalization performance of neural networks can be a difficult task due to the difficulty to find the best learning rate and momentum values. According to [28], if the learning rate is updated during the training process, the performance of the neural network can increase, achieving stable learning. Furthermore, for generalizing the MLP prediction, both the Bayesian regularization [28] and Levenberg-Marquardt [25] techniques may be applied, boosting the application of this neural network to achieve the aim of the present study.

B. Concepts of LSTM Neural Network

Ankle joint torque is a continuous signal that varies throughout the time and, thus, it can be recognized as a Time-Series signal. In this field, studies reported that the use of Deep Learning methods, such as LSTM, can provide satisfactory results [31], [32]. LSTM consists of a special type of recurrent neural networks. In the architecture of recurrent neural networks, the behavior of the activation functions of the hidden neurons depends not only on the behavior of the previous activation functions but also on the behavior at an earlier time. Thus, the recurrent neural networks can efficiently learn Time-Series data, since they consider that the behavior of the activation functions is dependent on the time [24]. LSTM is a conjugation of recurrent neural networks with gradient descendent learning algorithms, making easier to obtain good results with unstable gradient problems under control [29]. As reported to MLP, if only one learning rate and momentum value are applied, the weight update is the same in the neural network, causing the convergence performance less efficient. To improve the convergence performance, some gradient descendent algorithms have been emerged, such as, the Adaptive Moment Estimation (ADAM). With this algorithm, the weights of the neural network are updated using adaptive learning rates, avoiding the existence of a global one. Thus, the convergence performance is increased, while the learning error is reduced, obtaining a versatile neural network with the potential to deal with the aim of this paper [30].

III. Experimental Data Acquisition

In order to train and test each one of the implemented AI models, we performed a data acquisition, collecting kinematic and kinetic data from the lower limb joints of healthy subjects, walking at different speeds.

A. Participants

The data acquisition involved sixteen adult subjects (8 males and 8 females with a mean age of 23.8 ± 2.02 years, mean weight of 67.5 ± 10.8 kg and mean height of 1.69 ± 0.109 m), with no evidence of any type of physical and physiological disorder that could interfere with their walking pattern. The minimum and maximum body height registered was 1.51 and 1.83 m, whereas

Authorized licensed use limited to: Universita Studi di Torino - Dipartimento Di Informatica. Downloaded on September 12,2020 at 09:57:59 UTC from IEEE Xplore. Restrictions apply.

the minimum and maximum body mass was 52.0 and 83.7 kg, covering subjects of both genders with a wide range of body height and mass.

B. Instrumentation and Protocol

Although ankle joint is the focus of the present study, 3D joint kinematic and kinetic data of the remaining lower limb joints were also acquired for future developments at these joints. Thus, a motion-capture system with twelve cameras (Oqus; Qualysis – Motion Capture System, Göteborg, Sweden) and five force platforms embedded in the floor (FP4060; Bertec, Ohio, United States of America) were used to obtain the kinematic and ground reaction force (GRF) data at 200 Hz, respectively.

To determine the kinematic data, a Newington-Hayes marker set was adopted, integrating four more reflective markers positioned in the first metatarsal head, medial malleolus, medial tuberosity of the femur and in the trochanter, in order to achieve better measurements [31]. Thus, 24 reflective markers were used, as presented in Fig. 1.

Figure 1. Marker set adopted.

After the placement of the reflective markers, all subjects were asked to perform a standing static calibration trial to fit the anthropometric data to the body model of the acquisition system. Then, all subjects were instructed to perform ten walking trials at seven different walking speeds (1.0, 1.5, 2.0, 2.5, 3.0, 3.5 and 4 km/h) on a flat surface with 10 m, containing five embedded force platforms. Note that all the walking procedures were performed with the same and appropriate type of shoes to do not affect the walking motion dynamics.

C. Data Processing

The kinematic data acquired with the motion-capture system were low pass filtered with a zero-phase fourth-order Butterworth. The cutoff frequency chosen was 6 Hz. The GRF data acquired with the force platforms were filtered with the same filter, using a cutoff frequency of 10 Hz. The GRF was processed in *Visual3D* software to determine the torque values of all lower limb joints. These torque data were used as ground truth to the torque estimated by each implemented model.

IV. Implementation of Regression Models

A. Data Preparation

The collected ankle joint data was used to train, validate and test both proposed regression models. The input features of MLP and LSTM neural networks were the ankle joint angles, angular velocity, angular acceleration, body height, and walking speed. The body mass was used to normalize the output feature of the models: the ankle joint torque.

To train and to analyze the effectiveness of both models, the data were randomly divided: 60 % for training, 20 % for validation and 20 % for testing, corresponding to 10, 3 and 3 subjects, respectively. To analyze the robustness of both models, a *k*-fold cross-validation algorithm was implemented, where the number of folds (*k*) was set to 4.

We normalized the input and the output data to improve the learning process, achieving the model convergence with faster computations times. For this purpose, we used the median normalization method, described in Equation (1), where X represents the variable to normalize and X_{norm} corresponds to the normalized variable.

$$X_{norm} = \frac{X - \text{median}(X)}{\text{Interquartile Range}} \quad (1)$$

B. Implementation of MLP Regression Model

Using the neural network toolbox of *MATLAB R2018b*, an MLP with backpropagation algorithm was applied, investigating three different training algorithms: **(*i*)** *trainbr*, corresponding to the Bayesian regularization method; **(*ii*)** *trainlm*, corresponding to Levenberg-Marquardt method; and **(*iii*)** *traingdx*, to update the learning rate parameter during the training process of the neural network. Note that, in the neural network toolbox of *MATLAB R2018b* is not possible to apply the ADAM algorithm when applying an MLP neural network.

The main characteristics imposed during the training process are presented in Table I, where the parameters were defined according to the default values presented in the toolbox of *MATLAB R2018b* and explained in [32]. The stopping criteria were based on the maximum number of epochs, maximum validation failures, minimum performance gradient or maximum momentum.

TABLE I. PARAMETERS FOR MLP

Parameter	Value
Maximum number of epochs	10000
Performance goal	0
Maximum validation failures	10
Minimum performance gradient	$1e^{-7}$
Initial *mu*	0.001
Momentum decrease factor	0.1
Momentum increase factor	10
Maximum Momentum	$1e^{10}$

Authorized licensed use limited to: Universita Studi di Torino - Dipartimento Di Informatica. Downloaded on September 12,2020 at 09:57:59 UTC from IEEE Xplore. Restrictions apply.

Since there are no related studies in the literature, the architecture of the MLP was investigated regarding **(*i*)** the number of neurons (NN), considering 10, 70 and 110; and **(*ii*)** the number of hidden layers (NHL) varying from 1 to 2.

C. Implementation of LSTM Regression Model

Table II presents the main options defined during the training process of LSTM, according to the default values of the neural network toolbox of *MATLAB R2018b*. The neural network trained while the number of maximum validation failures or the number of maximum epochs was not reached. Moreover, we explored different NN in LSTM, varying between 10, 70 and 110, to find the structure that provides the best predictions.

TABLE II. PARAMETERS FOR LSTM

Parameter	Value
Maximum number of epochs	10000
Maximum validation failures	10
Gradient Descendent Algorithm	ADAM
Initial Learning Rate	0.01
Learning Rate Drop Period	50
Learning Rate Drop Factor	0.2
Batch Size	64

D. Model Evaluation Metrics

To evaluate the performance of the regression models, two metrics were used: Goodness of Fit (GOF) and Normalized Root Mean Square Error (NRMSE) between the predicted and the real ankle joint torque. These evaluation metrics were chosen since **(*i*)** the GOF evaluates the performance of the fit achieved with each model, where $-\infty$ represents a very poor fit and 100 % corresponds to a perfect fit; and **(*ii*)** the NRMSE provides a percentage of the Root Mean Square in the range of the experimental data.

The GOF is determined by Equation (2):

$$\mathrm{GOF} = \left(1 - \left(\frac{\|y_{ref} - y\|}{\|y_{ref} - \overline{y_{ref}}\|}\right)\right) \times 100 \qquad (2)$$

Where y_{ref} symbolizes the real parameter (real ankle joint torque), $\overline{y_{ref}}$, represents the mean of the real parameter and y is the estimated parameter (estimated ankle joint torque).

The NRMSE was computed as described in Equation (3), where $max_{y_{ref}}$-$min_{y_{ref}}$ represents the range of the real ankle joint torque:

$$\mathrm{NRMSE} = \frac{\mathrm{RMSE}}{\max_{y_{ref}} - \min_{y_{ref}}} \times 100 \qquad (3)$$

V. RESULTS AND DISCUSSION

A. MLP Regression Model

Table III presents the results obtained for the MLP neural network, considering different training functions and different architectures.

TABLE III. MLP RESULTS

Method	NHL	NN	GOF (%)	NRMSE (%)
MLP (*trainlm*)	1	10	55.3	9.41
		70	66.8	7.00
		110	69.0	6.54
	2	10	67.8	6.78
		70	69.4	6.44
		110	69.8	6.36
MLP (*trainbr*)	1	10	58.8	8.67
		70	69.1	6.50
		110	69.9	6.33
	2	10	67.0	6.95
		70	**70.9**	**6.13**
		110	69.2	6.50
MLP (*traingdx*)	1	10	46.9	11.2
		70	36.8	13.3
		110	32.5	14.2
	2	10	45.1	11.6
		70	40.5	12.5
		110	28.3	15.1

Based on the results achieved with *trainlm* and *trainbr* training functions, the regression performance was very similar. An increment in the NN improves the fit, achieving **(*i*)** a GOF's increase around 12.0 % and an NRMSE's decrease around 2.60 %, when 1 hidden layer is considered; **(*ii*)** an increment around 3.75 % of the GOF and a decrement around 0.620 % of the NRMSE when 2 hidden layers are considered. Note that these percentages correspond to the average values obtained by the GOF's increase and NRMSE's decrease concerning both *trainlm* and *trainbr* training functions. Additionally, an increase in the NHL was reflected by an improvement in the regression performance. However, considering 2 hidden layers, it was noticed that this performance's improvement became less appreciable when the NN was incremented. Using *traingdx* as the training function, it would be expected to achieve better performances, considering that an adaptive learning rate during the training process produces better results [28]. However, with this training function, the worst results were achieved (GOF $\leq$ 46.9 % and NRMSE $\geq$ 11.2 %). Thus, this training function is the least indicated to predict the ankle joint torque.

Overall, findings of Table III indicate that an MLP with 2 hidden layers, 70 hidden neurons and trained with *trainbr* as

Authorized licensed use limited to: Universita Studi di Torino - Dipartimento Di Informatica. Downloaded on September 12,2020 at 09:57:59 UTC from IEEE Xplore. Restrictions apply.

training function was the architecture that provided the best performance to the predict the reference ankle joint torque.

B. LSTM Regression Model

The results achieved for the LSTM regression model can be consulted in Table IV.

TABLE IV. LSTM RESULTS

Batch Size	NN	GOF (%)	NRMSE (%)
64	10	77.7	4.70
	70	78.4	4.55
	110	**79.6**	**4.31**

Based on the achieved results, we verified that an increment in the NN did not offer considerable improvements in the ankle joint torque prediction, being the best performances reached when 110 neurons are considered (achieving a GOF of 79.6 % and an NRMSE of 4.31 %).

In this neural network, the effect of the batch size was studied. However, due to the high time consumption encountered during the neural network's training, the batch size was only varied for the LSTM with 110 neurons, since that was the architecture that provided the best performance considering the default parameters presented in Table IV. The results of this variation are exhibited in Table V.

TABLE V. LSTM RESULTS WITH BATCH SIZE VARIATION

Batch Size	NN	GOF (%)	NRMSE (%)
128	110	79.5	4.57
64	**110**	**79.6**	**4.31**
32	110	76.8	4.87

According to Table V, as the batch size increases, similar performances are obtained (achieving GOF and NRMSE values of 79.5 % and 4.57 %, respectively). On the other side, as the batch size decreases, the performance of the LSTM started to decline, presenting a reduction of 2.80 % in the GOF and an increase of 0.560 % in the NRMSE values. Thus, an LSTM with 110 neurons and trained with a batch size of 64 was found as the architecture that provided the best performances.

C. Comparative Analysis of Regression Models

Considering the best results achieved for MLP and LSTM neural networks (presented in Tables III and V, respectively), it is possible to infer that LSTM was considerably better than MLP, exhibiting higher GOF and lower NRMSE values (GOF: 79.6 % > 70.9 %; NRMSE: 4.31 % < 6.13 %). The reliability of these results can be confirmed by Fig. 2, where it is possible to conclude that the LSTM generated reference ankle joint torques closer to the expected ones when compared to the predictions made by MLP. By performing a deeper analysis centered on the gait cycle, during the stance phase, it was verified that the MLP was also able to achieve an adequate estimation of the ankle joint torque. However, in the swing phase, irregularities and sudden peaks were verified. On the other hand, LSTM was able to fit the ankle joint torque throughout the full gait cycle. This fact can be justified by the architecture of this neural network, which makes it able to consider present and past information, offering good performances for time-series data, such as the ankle joint torque. The findings of this work showed the reliability of AI regression models to achieve accurate ankle joint torque predictions. Particularly, the implemented LSTM model has the potential to be integrated into control architectures of the robotic assistive devices to accurately estimate user-oriented reference joint torque trajectories needed in AAN control strategies. To the best knowledge of the authors, there is no previous literature work focused on the ankle joint torque trajectory generation based on well-known data from each subject, avoiding prior data collections. Thus, it is not possible to perform a comparative analysis with literature algorithms.

D. Limitations

Results show that the LSTM was able to estimate healthy user-oriented ankle joint torque trajectories. However, the generation of these trajectories is only valid when: ***(i)*** the walking speed is between 1.0 and 4.0 km/h; ***(ii)*** the subject's

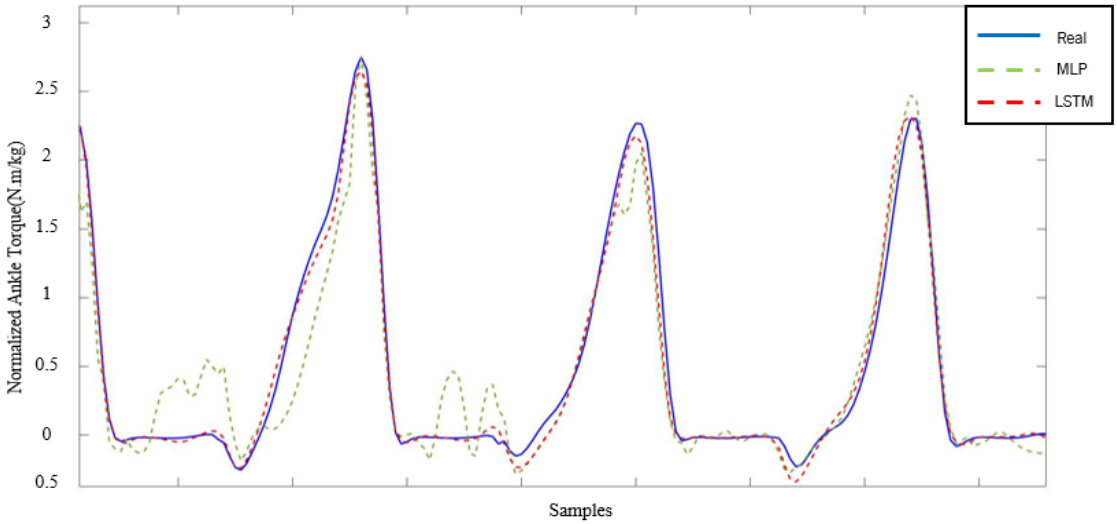

Figure 2. Predictions of the MLP and LSTM models in comparison to the real ankle joint torque for a random selected range of the test dataset.

Authorized licensed use limited to: Universita Studi di Torino - Dipartimento Di Informatica. Downloaded on September 12,2020 at 09:57:59 UTC from IEEE Xplore. Restrictions apply.

body height ranges from 1.51 to 1.83 m; ***(iii)*** the subject's body mass is higher than 52.0 and lower than 83.7 kg; and ***(iv)*** the locomotion mode is on a flat surface.

VI. Conclusions and Future Perspectives

The accurate prediction of healthy reference ankle joint torques oriented to the user is of utmost importance on AAN control strategies that require an adequate healthy reference torque in their architecture towards personalized robotic-based gait assistance. Based on the obtained results, it is possible to conclude that the ankle joint torque can be accurately generated by an LSTM for subjects with a body height ranging from 1.51 to 1.83 m, a body mass between 52.0 and 83.7 kg and walking on a flat surface with a walking speed ranging from 1.0 to 4.0 km/h.

Future challenges involve the determination of the joint torque for the remaining lower limb joints, namely, knee and hip. On the other hand, the exploration of other machine learning algorithms could be issued to improve the results achieved in the present study. Furthermore, to address the limitations of this study, future challenges also involve the generation of reference joint torque trajectories for a wider range of body masses, heights, and walking speeds for different locomotion modes, such as ramps or stairs.

Acknowledgment

We thank Pedro Fonseca for the assistance in data collection in LABIOMEP - Porto Biomechanics Laboratory, University of Porto, Porto, Portugal.

References

[1] S. H. Jang, "The recovery of walking in stroke patients: a review," *Int. J. Rehabil. Res.*, vol. 33, no. 4, pp. 285–289, Dec. 2010.

[2] D. G. da Saúde, "Programa Nacional de Prevenção e Controlo das Doenças Cardiovasculares," *Dgs*, p. 28, 2006.

[3] W. Meng, Q. Liu, Z. Zhou, Q. Ai, B. Sheng, and S. S. Xie, "Recent development of mechanisms and control strategies for robot-assisted lower limb rehabilitation," *Mechatronics*, vol. 31, pp. 132–145, Oct. 2015.

[4] P. K. Jamwal, S. Q. Xie, S. Hussain, and J. G. Parsons, "An Adaptive Wearable Parallel Robot for the Treatment of Ankle Injuries," *IEEE/ASME Trans. Mechatronics*, vol. 19, no. 1, pp. 64–75, Feb. 2014.

[5] J. A. Saglia, N. G. Tsagarakis, J. S. Dai, and D. G. Caldwell, "A High-performance Redundantly Actuated Parallel Mechanism for Ankle Rehabilitation," *Int. J. Rob. Res.*, vol. 28, no. 9, pp. 1216–1227, Sep. 2009.

[6] G. Colombo, M. Joerg, R. Schreier, and V. Dietz, "Treadmill training of paraplegic patients using a robotic orthosis.," *J. Rehabil. Res. Dev.*, vol. 37, no. 6, pp. 693–700, 2000.

[7] D. Aoyagi, W. E. Ichinose, S. J. Harkema, D. J. Reinkensmeyer, and J. E. Bobrow, "A Robot and Control Algorithm That Can Synchronously Assist in Naturalistic Motion During Body-Weight-Supported Gait Training Following Neurologic Injury," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 15, no. 3, pp. 387–400, Sep. 2007.

[8] C. A. Fukuchi, R. K. Fukuchi, and M. Duarte, "A public dataset of overground and treadmill walking kinematics and kinetics in healthy individuals," *PeerJ*, vol. 2018, no. 4, pp. 1–17, 2018.

[9] B. Koopman, E. H. F. van Asseldonk, and H. van der Kooij, "Speed-dependent reference joint trajectory generation for robotic gait support," *J. Biomech.*, vol. 47, no. 6, pp. 1447–1458, Apr. 2014.

[10] Y. He *et al.*, "GC-IGTG: A Rehabilitation Gait Trajectory Generation Algorithm for Lower Extremity Exoskeleton," in *2019 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2019, no. 2017, pp. 2031–2036.

[11] C. Fleischer and G. Hommel, "Calibration of an EMG-Based Body Model with six Muscles to control a Leg Exoskeleton," in *IEEE International Conference on Robotics and Automation*, 2007, pp. 2514–2519.

[12] C. Fleischer and G. Hommel, "A Human-Exoskeleton Interface Utilizing Electromyography," *IEEE Trans. Robot.*, vol. 24, no. 4, pp. 872–882, Aug. 2008.

[13] W. Hassani, S. Mohammed, and Y. Amirat, "Real-Time EMG Driven Lower Limb Actuated Orthosis for Assistance As Needed Movement Strategy," in *Robotics: Science and Systems IX*, 2013.

[14] W. Hassani, S. Mohammed, H. Rifaï, and Y. Amirat, "Powered orthosis for lower limb movements assistance and rehabilitation," *Control Eng. Pract.*, vol. 26, pp. 245–253, May 2014.

[15] P. Artemiadis, "EMG-based Robot Control Interfaces: Past, Present and Future," *Adv. Robot. Autom.*, vol. 1, no. 2, pp. 1–3, 2012.

[16] R. M. Singh, S. Chatterji, and A. Kumar, "Trends and challenges in EMG based control scheme of exoskeleton robots—a review," *Int. J. Sci. Eng. Res*, vol. 3, no. 8, pp. 933–940, 2012.

[17] J. Zhang, C. C. Cheah, and S. H. Collins, "Experimental comparison of torque control methods on an ankle exoskeleton during human walking," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, 2015.

[18] J. S. Campos Figueiredo, "Smart Wearable Orthosis to Assist Impaired Human Walking," University of Minho, 2019.

[19] D. T. H. Lai, P. Levinger, R. K. Begg, W. L. Gilleard, and M. Palaniswami, "Automatic Recognition of Gait Patterns Exhibiting Patellofemoral Pain Syndrome Using a Support Vector Machine Approach," *IEEE Trans. Inf. Technol. Biomed.*, vol. 13, no. 5, pp. 810–817, Sep. 2009.

[20] M. Yang, H. Zheng, H. Wang, S. McClean, J. Hall, and N. Harris, "A machine learning approach to assessing gait patterns for Complex Regional Pain Syndrome," *Med. Eng. Phys.*, vol. 34, no. 6, pp. 740–746, Jul. 2012.

[21] H. Asadi, R. Dowling, B. Yan, and P. Mitchell, "Machine Learning for Outcome Prediction of Acute Ischemic Stroke Post Intra-Arterial Therapy," *PLoS One*, vol. 9, no. 2, pp. 1–11, Feb. 2014.

[22] R. Caldas, T. Fadel, F. Buarque, and B. Markert, "Adaptive predictive systems applied to gait analysis: A systematic review," *Gait Posture*, vol. 77, no. May 2019, pp. 75–82, Mar. 2020.

[23] C. Lv *et al.*, "Levenberg–Marquardt Backpropagation Training of Multilayer Neural Networks for State Estimation of a Safety-Critical Cyber-Physical System," *IEEE Trans. Ind. Informatics*, vol. 14, no. 8, Aug. 2018.

[24] M. Nielsen, "Neural Networks and Deep Learning," in *Artificial Intelligence*, 2015.

[25] M. Kayri, "Predictive Abilities of Bayesian Regularization and Levenberg–Marquardt Algorithms in Artificial Neural Networks: A Comparative Empirical Study on Social Data," *Math. Comput. Appl.*, vol. 21, no. 2, p. 20, May 2016.

[26] M. T. Hagan and M. B. Menhaj, "Brief Papers," *Brain Cogn.*, vol. 32, no. 2, pp. 273–344, Nov. 1996.

[27] P. N. Fernandes *et al.*, "EMG-based Motion Intention Recognition for Controlling a Powered Knee Orthosis," in *2019 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, 2019, pp. 1–6.

[28] H. Demuth, M. Beale, and M. Hagan, "Neural network toolbox™ 6," *User's Guid.*, 2008.

[29] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Comput.*, vol. 9, no. 8, Nov. 1997.

[30] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," pp. 1–15, Dec. 2014.

[31] M. P. Kadaba, H. K. Ramakrishnan, and M. E. Wootten, "Measurement of Lower Extremity Kinematics During Level Walking," *J. Orthop. Res.*, vol. 8, pp. 383–392, 1990.

[32] J. F. Mas and J. J. Flores, "The application of artificial neural networks to the analysis of remotely sensed data," *Int. J. Remote Sens.*, vol. 29, no. 3, pp. 617–663, Feb. 2008.

Authorized licensed use limited to: Universita Studi di Torino - Dipartimento Di Informatica. Downloaded on September 12,2020 at 09:57:59 UTC from IEEE Xplore. Restrictions apply.