2023 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)
June 28-30, 2023. Seattle, Washington, USA

Prediction Accuracy and Model Robustness of Neural Network-Based Ground Reaction Force Estimators

Mohamed Abdelhady[1,2], Thomas C. Bulea[1], Wael Abouelwafa[3] and Dan Simon[2]

***Abstract*—Ground reaction force (GRF) is a potentially useful control input for powered lower limb prostheses but accurate GRF measurement in real-time is challenging. The objective of this work is to evaluate the ability to estimate GRF from a minimal set of kinematic inputs (knee and ankle angle and angular velocities) during walking. Three artificial neural networks (ANNs) are evaluated for this purpose: nonlinear autoregression with exogenous input (NARX), delayed discrete recurrent neural network (DDRNN), and a self-organizing map with feedforward neural network (SOM-FFNN). Specifically, our work focuses on investigating the impact of ANN architecture and training/learning algorithms on the predication accuracy of GRF. First, ANN performance in open loop GRF estimation is investigated using treadmill walking data in a healthy participant at speeds from 0.79 to 1.9 m/s. Next, the effect of ANN estimated GRF is evaluated in a simulation of closed-loop powered prosthesis control with three levels of measurement noise. The results show that all ANNs are able to estimate GRF in open-loop with relatively low RMS, although SOM-FFNN performed the best with an average RMS of 4.85 N across all gait speeds. SOM-FFNN also showed the most robust performance in estimating GRF for trajectory tracking in closed-loop control, providing impetus for its further investigation in control of powered prostheses.**

***Index Terms*—Ground reaction force, self-organized map, neural network, recurrent network, estimation.**

I. Introduction

There are at least 1.6 million people living in the U.S. with an amputation, of which more than 50% are of a lower extremity and this number is expected to more than double by the year 2050 [1]. Thus, there has been considerable effort to develop more effective lower limb prostheses to restore walking function in these individuals. In particular, recent advances have focused on developing powered prostheses with more robust feedback control to enable use in a variety of walking tasks and environments [2], [3].

[1]Mohamed Abdelhady and Thomas C. Bulea are with the Neurorehabilitation and Biomechanics Research Section, Rehabilitation Medicine Department, National Institutes of Health Clinical, Bethesda, MD, 20892 USA. mohamed.abdelhadi@nih.gov; thomas.bulea@nih.gov

[2]Dan Simon is with Department of Electrical and Computer Engineering Dept., Cleveland State University, Cleveland, OH 44115, USA. d.j.simon@csuohio.edu.

[3]Wael Abuoelwafa is with Biomedical Engineering Department, Minia University, Minia, Egypt. wael.wafa@mu.edu.eg.

This work was supported by the Intramural Research Program of the NIH Clinical Center.

Ground reaction force (GRF) is the force exerted by the ground on a body in contact with it. GRF has long been an essential component for gait analysis because it implicitly includes physiological information and motion intent [4]. For example, GRF was reported as a potential tool for the early diagnosis and monitoring of Parkinson's disease [5], [6] and as a determinant of gait function in individuals with amyotrophic lateral sclerosis (ALS) [7], Huntington's disease [8], and stroke [9].

GRF is also attractive as a control input for powered prostheses to adapt joint stiffness and damping to mimic natural walking across a variety of terrains [10], [11]. Measuring GRF in real-time presents multiple challenges. First, human walking (and GRF) can vary widely depending on the individual's weight, walking speed, and surface conditions; thus a high bandwidth measurement system with multiple sensors is required, which can increase system complexity [12]. Next, relatively small changes in GRF can indicate large changes in gait biomechanics, which necessitates high-precision and high-resolution equipment. Finally, measurement systems must process and output the data quickly for real-time applications such as prosthesis control or biofeedback. Given the complexity required to accurately measure GRF directly, interest in GRF estimation has increased [13]. If a GRF estimator can provide a reasonable approximation, it may be more efficiently implemented in time-critical applications such as real-time control. GRF estimation accuracy is crucial to ensure user safety and achieve optimal prosthesis performance across a variety of tasks. Therefore, GRF estimation algorithms should be designed for high accuracy and robustness to sources of error, such as sensor noise, measurement drift, and model uncertainties.

Neural networks are particularly well-suited for estimation tasks because they can learn complex, non-linear relationships between inputs and outputs even when the underlying relationships are not fully understood [13]. Thus, neural networks can be trained on a set of input-output pairs and then make predictions for new inputs not seen during training. Neural networks can also handle large amounts of data and high-dimensional input spaces and are able to learn from noisy or incomplete data, which is often the case in real-world applications. These characteristics make them well-suited for sensor-based estimation tasks where the data is uncertain or unreliable.

Artificial neural networks (ANNs) have been used to estimate GRF from other signals, such as kinematics, electromyography (EMG), or data from inertial measurement units (IMUs) [14], [15], [16], [17]. Collectively, these studies demonstrate the ability of ANNs to estimate GRF from a variety of inputs, while also highlighting key factors that affect ANN accuracy including the number of sensors and their position on the body and the number and type of ANN inputs. The training data and its cross-correlation is another important factor for ANNs; training data should be selected to include non-correlated inputs. [18] estimated GRF during running based on three inertial sensors on the shank and pelvis, which produced correlated inputs. The ANN architecture was developed to mitigate the data dependency problem by independently training multiple ANNs. Learning style is another important factor for ANNs. Supervised learning is more straightforward than unsupervised learning but unsupervised learning is more robust. [19] present a self-organizing ANN for the recognition of human-object interactions. They applied an ANN called grow-when-required that showed ANN architecture and learning style played a vital role in performance.

This study investigates a novel approach to GRF estimation using ANN, with the ultimate goal of using GRF for control of powered lower limb prostheses and orthoses. This study includes three main contributions. First, whereas previous studies have estimated GRF from a large number of measurements [11], [15], we use a set of easily obtained inputs, knee and ankle angular displacement and velocity, which reduces computational complexity. Second, we evaluate three ANN architectures for GRF estimation from this reduced input set: nonlinear autoregressive network with exogenous inputs (NARX), discrete delayed recurrent neural network (DDRNN), and a novel approach utilizing a feed foward neural network with a self organizing map (SOM-FFNN). Third, we evaluate performance of each ANN in a robotic prosthesis simulation to assess performance in closed-loop control.

II. METHODS

The data in this study come from the Louis Stokes Cleveland Veteran Affairs Medical Center (VAMC). Motion data was captured from a single able-bodied subject, after informed consent, with weight 78 kg and height 180 cm. A Vicon system with 16 cameras recorded lower extremity joint angles at 100 Hz. GRF was recorded using force plates under the treadmill at 100 Hz to provide ground truth GRF data [20]. Data were recorded at walking speeds of 0.75, 0.8, 1.25, 1.3, 1.44, 1.63, and 1.9 m/s. Each data set has at least 4200 samples per trial, and each walking speed includes at least four trials. Training was based on a single trial at 1.3 m/s, and the

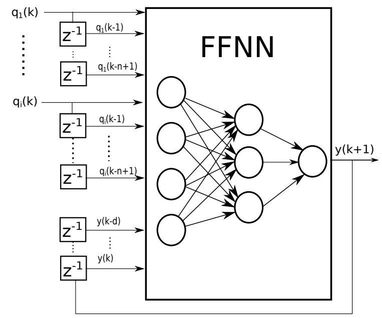


Fig. 1: NARX network architecture with *n+1* delayed inputs and *d* delayed outputs (z^{-1} is the unit time delay).

rest of the data are used for testing and validation. The ANN inputs are knee angle q_K, knee angular velocity $\dot{q}_K$, ankle angle q_A, and ankle velocity $\dot{q}_A$.

Three ANN architectures were applied to estimate GRF. NARX is well-suited to periodic signals such as human gait, to nonlinear relations between input and output, and often exhibits good robustness to input noise [21]. DDRNN is a new recursive network that implements associative memory and state space representation [22], [23], [24]. SOM-FFNN is introduced in this paper for the first time to estimate GRF. This novel approach infers hidden features from the data with an SOM based on gait speed, and then those features are input to an FFNN to estimate GRF.

A. NARX Architecture

NARX is a recurrent dynamic network with feedback connections enclosing several layers of the network. The NARX model is based on the linear ARX model, which is commonly used in time-series modeling. NARX estimates a nonlinear regressive model of order n based on measured data [25]. The general equation that represents a NARX model is

$$\begin{aligned} y(k+1) = f\Big(& y(k), \dots, y(k-d), q_1(k), \dots, \\ & q_i(k), \dots q_i(k-n+1)\Big) + \epsilon(k) \end{aligned} \tag{1}$$

where $n \in \mathbb{Z}^+$ is the regression order, q_i are the inputs, y_i are the outputs, $f(\cdot)$ is a nonlinear mapping, and $\epsilon(k)$ is the modeling residual.

NARX time series modeling is well-known in the system identification literature and is commonly implemented with a neural network. The NARX architecture used in this paper is illustrated in Fig. 1, where the inputs to the network are the number of exogenous signals given by i , the number of previous output samples given by d, and the number of time samples of each of the i exogenous signals given by n.

In our implementation, input-layer neurons were linear and the hidden and output-layer neurons were $tansig$

(tangent-sigmoid) functions [26]. NARX models are typically trained with the Levenberg-Marquardt algorithm, but other studies recommend backpropagation through time (BPTT) [27]. Both algorithms were investigated here, with LM implented as in [28] and BPTT implemented as in [26], [29], [30]. Data from walking at 1.3 m/s was used to train NARX using both BPTT and LM. The data was partitioned to use 60% for training, 30% for local cross validation, and 10% for local-testing. Data from the other walking speeds were used for testing.

NARX includes 25 input-layer neurons: 5 for each input (knee angle, knee velocity, ankle angle, and ankle velocity) and 5 for output feedback. That is, $i = 4$ and $n = d = 5$ in Fig. 1. The hidden layer has 10 neurons and the output layer has 1. The activation function for the input and output layers are linear, and the activation function for the hidden layer is $tansig$.

The training with LM (R=0.99) performed better than BPTT (R=0.97) for NARX. Varying the delay parameter d, which represents the number of previous outputs used to estimate the next GRF, directly affected NARX performance. GRF predictions were concave with respect to parameter d, indicating the optimal value was $d = 10$.

B. DDRNN Architecture

DDRNN is a modification of NARX with the ability to handle additional model complexity and nonlinearities through recurrence [22], [23], [30]. DDRNN can be represented in a flexible canonical state-space form comprising a feedforward ANN **N**, a multi-delay element **D**, a nonlinear mapping function Φ, and a unit delay element $\mathbf{z}^{-1}I$ (Fig. 2). Here, $x \in \mathbb{R}^n$ is the state vector of the ANN, and $\zeta(\cdot)$ is the input vector to the nonlinear operator ϕ. The feedforward ANN, **N**, is implicitly treated as a linear mapping.

The RNN architecture is appealing because analyzing the RNN is more straightforward than NARX [22] and DDRNN stability conditions, which provide learning constraints, can be easily derived. The stability proof of DDRNN has been previously published; exponential as well as asymptotic stability can be proven using Lyapunov-Krasovkii stability theory [24], [31].

$\Phi(\cdot)$ is chosen to be the $tanh$ (hyperbolic-tangent) function. **N** is initialized to a matrix with the identity matrix as its leftmost component and zeros on its right side. The input vector is q_k, $\dot{q}_k$, q_A, and $\dot{q}_A$ and the output y is the estimated GRF. BPTT was used to train DDRNN with data at walking speed 1.3 m/s with the same partitions as NARX. The remainder of the data are used for testing. The training and validation outcomes were R=0.96 and R=0.97 respectively for DDRNN.

C. Feed Forward Self Organized Map Architecture

SOM-FFNN combines the classification ability of an SOM with a feedforward ANN. The SOM is comprised of an $r \times r$ array of neurons with input vector $\mathbf{q} = (q_1, \cdots, q_n)$ and m outputs that are input to the FFNN (Fig. 3). Here, the SOM classifies walking speed based on knee and ankle angles and velocities before the FNN estimates GRF. The SOM is an ANN trained with unsupervised learning to map inputs from a high-dimensional space to a low-dimensional space, or vice versa [32]. Combining FFNN with SOM as a preprocessing unit has been used before in other applications, where a SOM was used to extract features from a high dimensional space and pass them to a k-NN neural network [33].

The advantages of SOM are its competitive rather than error-correction learning and its neighborhood function, which maintains topological input characteristics. The SOM structure is an array of $r \times r$ neurons arranged in a two-dimensional lattice, termed the competitive neuron layer, with a specific topology. Here, a square topology is used and each neuron in the competitive layer has weights $\mathbf{w}_{ij} \in \mathbb{R}^n$, where n is the number of inputs and $i, j \in [1, r]$. When input vector $\mathbf{q}$ is presented to the network, the lattice neurons compete and the winning neuron index is denoted as $index$. Note that $index$ includes an (i, j) pair. $\mathbf{w}_{index}$ indicates the neuron weights that are most similar to the input vector $\mathbf{q}$.

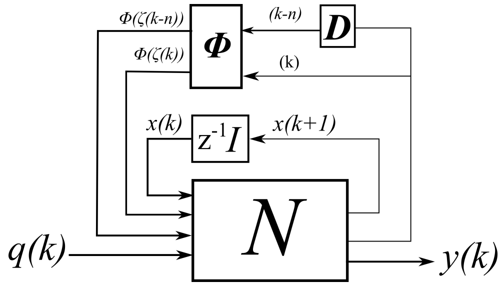


Fig. 2: Discrete delayed recurrent neural network architecture.

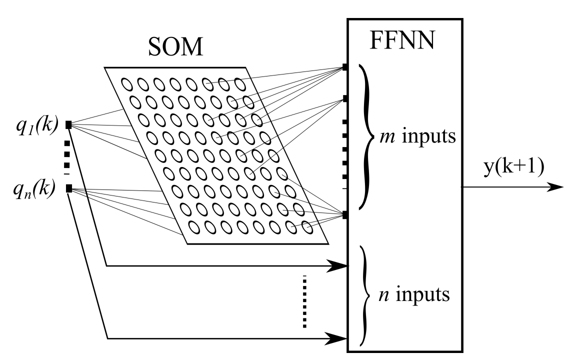
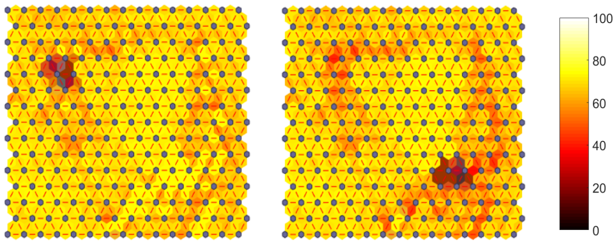


Fig. 3: SOM-FFNN estimator composed of self-organizing map and feedforward ANN.



(a) (b)

Fig. 4: The SOM for (a) 1.63 m/s and (b) 1.3 m/s. The connection between neurons is visualized through color coding, where a dark color indicates a strong relationship and a light color represents a weak relationship.

The winning neuron is known as the best matching unit (BMU). The BMU index is

$$index = \arg\min_{i,j} \|q(k) - w_{ij}\|, \quad i, j = 1, \dots, r \tag{2}$$

The BMU neighborhood radius is initialized to include the entire lattice and at each time step the neuron weights in the neighborhood radius are updated to be closer to the input vector:

$$w_{ij}(k+1) = w_{ij}(k) + \alpha(k)h(index, k)(q(k) - w_{ij}(k)) \tag{3}$$

where $0 < \alpha(k) < 1$ is the learning rate and $h(index, k)$ is a weighting function that limits the BMU neighborhood size. The weighting function used here is the Gaussian function

$$h(index, k) = \exp\left(-\frac{\|q(k) - w_{index}(k)\|^2}{2\sigma^2(k)}\right) \tag{4}$$

where $\sigma(k)$ is the neighborhood radius, $q(k)$ is the current input, and $w_{index}(k)$ is the weight of the current BMU. Each output represents a known category (for example, walking speed). The algorithm updates the neurons one at a time, then assigns each output to a specific category corresponding to input **q**. The SOM network is trained using learning vector quantization (LVQ) [34] via the Matlab deep learning toolbox [35].

Here, the SOM network is composed of a 15×15 grid with the input layer having four nodes, i.e., $w_{ij} \in \mathbb{R}^4$. The SOM training data included one stride from each walking speed. The rest of the data are used for testing and validation. The SOM produces seven distinct outputs for the seven different walking speeds, with an example shown in Figure 4 at two walking speeds (1.3 and 1.63m/s). For each walking speed, the SOM displays a set of substantial inter-neuron connections that classify the input into specific categories. There is some overlap between the groups, but the center of each group can be located on the diagonal of the SOM. These outputs are then fed into a FFNN.

For a fair comparison, the FFNN architecture was selected to be similar to that of the DDRNN and NARX including 11 input layer neurons (7 from the SOM, and 4 from the knee and ankle angles and velocities), 10 hidden layer neurons, and 1 output neuron (GRF estimate). The 1.3 m/s walking data for training, local validation and local testing the FFNN were partitioned the same as the NARX and DDRNN. After training SOM-FFNN, the correlation coefficient between actual and estimated GRF results from local-testing was R=0.997.

D. Validation in Closed-loop Prosthesis Control

One goal of this research is to estimate GRF to reduce uncertainty in prosthetic leg control. Thus, we compared the GRF estimators in a closed-loop robotic prosthesis simulation [10]. The robot emulates the vertical hip displacement and angular thigh rotation of a transfemoral amputee. A passivity-based controller developed in [36] provides robust tracking for the hip and thigh joints and robust impedance control for the prosthesis knee and ankle joints. A treadmill provides a dynamic walking surface. The reference data include 10 strides at walking speed of 1 m/s. The vertical and horizontal GRF components are calculated from the robot dynamics.

The control feedback signal includes vertical hip displacement and velocity, thigh angle and velocity, knee angle and velocity, ankle angle and velocity, and GRF as estimated by the ANN. Control performance was evaluated by measuring the deviation of the simulated trajectories from the reference trajectories, and is quantified by integral square error (ISE). Simulations were conducted with no, low (SNR = 6 dB), and high (SNR = 3.5 dB) noise.

III. Results

A. Validation Tests

The accuracy of each estimation method was judged by its success in predicting GRF from unseen data during a validation process. 10% of the 1.3 m/s data was used for local validation after the training process, while all other data was used for overall testing of GRF estimation. Table I shows the RMS value between estimate and actual GRF, as well as the validation results for each estimation technique. SOM-FFNN showed the lowest RMS error in GRF estimation across walking speeds while NARX showed the highest RMS. An example of the SOM-FFNN GRF estimate at 1.3 m/s for three consecutive gait cycles is shown in Fig. 5.

TABLE I: Local and Overall ANN Validation Results

Validation	NARX	DDRNN	SOM-FFNN
R_{Local}	0.91	0.95	0.996
$R_{Overall}$	0.99	0.98	0.99
RMS (N)	12.81	9.57	4.85

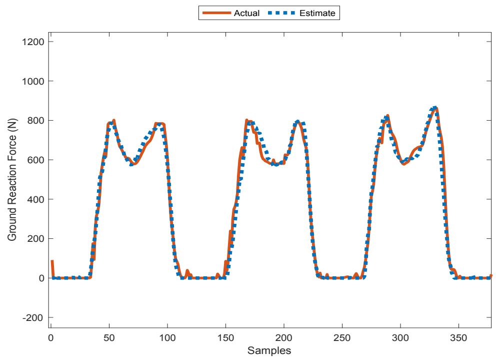


Fig. 5: SOM-FFNN estimate and actual ground reaction force at 1.3 m/s walking speed.

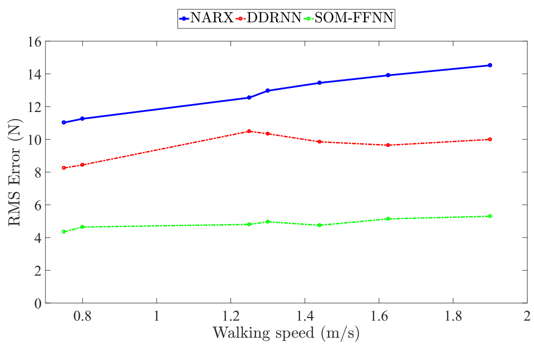


Fig. 6: Validation of SOM-FFNN, NARX, and DDRNN using all walking speed data.

B. Closed-Loop Control

As the aim of this study was to create a GRF estimator to reduce uncertainty and increase accuracy of torque calculation during closed loop control of lower limb prostheses, we evaluated each ANN in a simulated control environment. In the no-noise environment, control performance was similar for all estimation techniques (Table II), with the exception of better knee tracking performance for the NARX and better ankle tracking for SOM-FFNN. As expected, tracking errors for knee and ankle angle were worse than hip displacement and thigh angle due to the impedance control deployed at those joints, and their proximity to the application of GRF at the prosthetic foot which introduces uncertainty. GRF estimator-based control provided robust performance in both low and high noise simulations (Table II). As above, tracking performance was worse at the knee and ankle.

The difference between actual GRF from human motion data and ANN-estimated GRF is summarized in Table III. The table shows that the ANNs can provide accurate GRF estimates in a real-time simulation. SOM-FFNN gives better results than the other two ANN techniques. The performance of NARX is close to that of SOM-FFNN in the noise-free scenario. The relatively poor performance of DDRNN is likely due to its state-space architecture, which increases sensitivity to measurement noise in the closed-loop control environment.

IV. Discussion

Three different ANNs were investigated for GRF estimation with a minimal set of kinematic inputs available in a transfemoral prosthesis (knee and ankle angle and velocities). The GRF estimators were trained at walking speed of 1.3 m/s and their open-loop estimation performance was quantified at the remaining speeds from 0.79 to 1.9 m/s. The SOM-FFNN gave the best performance (average R=0.99 and average RMS=4.85 N) compared with DDRNN (R=0.98 and RMS=9.57 N) and NARX (R=0.99 and RMS=12.81 N).

Closed-loop GRF estimation performance was tested with a passivity-based simulated robotic prosthesis controller. The SOM-FFNN estimator was superior in closed-loop simulation. The NARX estimator performed second best. Whereas all GRF estimation techniques resulted in good tracking in the noise-free simulation, SOM-FFNN was more robust in the low-noise and high-noise scenarios.

In summary, nontraditional ANN estimator structures such as SOM-FFNN can provide accurate GRF estimation with a minimal number of kinematic inputs. The trade-off between SOM-FFNN architecture size and estimation accuracy is currently under investigation. Future work will focus on designing the SOM for use in a wider range of activities such as jumping, running, and climbing stairs, and then updating SOM-FFNN to include these additional activities, first in simulation and ultimately in a n experimental powered prosthesis. Ultimately, the integration of SOM-FFNN with closed loop techniques for trajectory tracking, such as impedance control, needs more study, especially in noisy environments. SOM-FFNN and other ANN GRF estimators may also be useful in providing biofeedback and other applications requiring real-time gait analysis.

References

[1] K. Ziegler-Graham, E. J. MacKenzie, P. L. Ephraim, T. G. Travison, and R. Brookmeyer, "Estimating the prevalence of limb loss in the united states: 2005 to 2050," *Archives of physical medicine and rehabilitation*, vol. 89, no. 3, pp. 422–429, 2008.

[2] M. R. Tucker, J. Olivier, A. Pagel, H. Bleuler, M. Bouri, O. Lambercy, J. d. R. Millán, R. Riener, H. Vallery, and R. Gassert, "Control strategies for active lower extremity prosthetics and orthotics: a review," *Journal of neuroengineering and rehabilitation*, vol. 12, no. 1, pp. 1–30, 2015.

[3] S. Rezazadeh, D. Quintero, N. Divekar, E. Reznick, L. Gray, and R. D. Gregg, "A phase variable approach for improved rhythmic and non-rhythmic control of a powered knee-ankle prosthesis," *IEEE Access*, vol. 7, pp. 109 840–109 855, 2019.

[4] M. Michałowska, T. Walczak, J. K. Grabski, and M. Cieślak, "People identification based on dynamic determinants of human gait," *Vibrations in Physical Systems*, vol. 29, 2018.

TABLE II: Tracking Error RMS With/Without Measurement Noise While Using GRF Estimates Compared with the Ground Truth of Using Actual GRF for Feedback Control

		NARX Estimate			DDRN Estimate			SOM-FFNN Estimate			Actual GRF		
		No noise	Low noise	High noise	No noise	Low noise	High noise	No noise	Low noise	High noise	No noise	Low noise	High noise
Hip position	(m)	0.27	0.52	0.73	0.24	0.31	0.63	0.23	0.24	0.52	0.37	0.21	0.65
Thigh angle	(deg)	1.03	4.17	9.85	1.23	6.31	9.29	1.25	4.35	4.85	1.63	5.02	3.48
Knee angle	(deg)	5.72	7.35	14.22	9.09	11.37	17.61	8.63	11.31	13.23	10.82	7.31	13.2
Ankle angle	(deg)	6.14	7.82	15.3	5.64	15.15	18.27	3.14	11.31	13.31	4.71	8.34	9.38

TABLE III: GRF RMS Estimation Error (N)

	No noise	Low noise	High noise
NARX	2.40	3.52	4.75
DDRNN	3.50	5.71	6.67
SOM-FNN	2.18	2.37	3.52

[5] M. N. Alam, A. Garg, T. T. K. Munia, R. Fazel-Rezai, and K. Tavakolian, "Vertical ground reaction force marker for parkinsons disease," *PloS one*, vol. 12, pp. 11–17, 2017.

[6] H. H. Manap, N. M. Tahir, and A. I. M. Yassin, "Statistical analysis of parkinson disease gait classification using artificial neural network," in *2011 IEEE International Symposium on Signal Processing and Information Technology*. IEEE, 2011, pp. 060–065.

[7] J. M. Hausdorff, A. Lertratanakul, M. E. Cudkowicz, A. L. Peterson, D. Kaliton, and A. L. Goldberger, "Dynamic markers of altered gait rhythm in amyotrophic lateral sclerosis," *Journal of Applied Physiology*, vol. 88, no. 6, pp. 2045–2053, 2000.

[8] J. M. Hausdorff, S. L. Mitchell, R. Firtion, C.-K. Peng, M. E. Cudkowicz, J. Y. Wei, and A. L. Goldberger, "Altered fractal dynamics of gait: reduced stride-interval correlations with aging and huntingtons disease," *Journal of Applied Physiology*, vol. 82, no. 1, pp. 262–269, 1997.

[9] E. Baratin, L. Sugavaneswaran, K. Umapathy, C. Ioana, and S. Krishnan, "Wavelet-based characterization of gait signal for neurological abnormalities," *Gait & Posture*, vol. 41, no. 2, pp. 634–639, 2015.

[10] H. Richter and D. Simon, "Robust tracking control of a prosthesis test robot," *Journal of Dynamic Systems, Measurement, and Control*, vol. 136, no. 3, p. 031011, 2014.

[11] E. C. Martinez-Villalpando, H. Herr, and M. Farrell, "Estimation of ground reaction force and zero moment point on a powered ankle-foot prosthesis," in 29^{th} *Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, 2007, pp. 4687–4692.

[12] M. A. Jamshed, K. Ali, Q. H. Abbasi, M. A. Imran, and M. Ur-Rehman, "Challenges, applications and future of wireless sensors in internet of things: A review," *IEEE Sensors Journal*, 2022.

[13] J. Lee, S. Sivakumar, and L. W. Ning, "A review on neural network based gait estimation methods," in *2022 International Conference on Green Energy, Computing and Sustainable Technology (GECOST)*. IEEE, 2022, pp. 358–364.

[14] A. Choi, J. M. Lee, and J. H. Mun, "Ground reaction forces predicted by using artificial neural network during asymmetric movements," *International Journal of Precision Engineering and Manufacturing*, vol. 14, no. 3, pp. 475–483, Mar 2013.

[15] S. E. Oh, A. Choi, and J. H. Mun, "Prediction of ground reaction forces during gait based on kinematics and a neural network model," *Journal of Biomechanics*, vol. 46, no. 14, pp. 2372–2380, 2013.

[16] Y. Guo, F. Storm, Y. Zhao, S. Billings, A. Pavic, C. Mazzà, and L.-Z. Guo, "A new proxy measurement algorithm with application to the estimation of vertical ground reaction forces using wearable sensors," *Sensors*, vol. 17, no. 10, p. 2181, 2017.

[17] M. Bataineh, T. Marler, K. Abdel-Malek, and J. Arora, "Neural network for dynamic human motion prediction," *Expert Systems with Applications*, vol. 48, pp. 26–34, 2016.

[18] F. J. Wouda, M. Giuberti, G. Bellusci, E. Maartens, J. Reenalda, B.-J. F. Van Beijnum, and P. H. Veltink, "Estimation of vertical ground reaction forces and sagittal knee kinematics during running using three inertial sensors," *Frontiers in Physiology*, vol. 9, p. 218, 2018.

[19] L. Mici, G. I. Parisi, and S. Wermter, "A self-organizing neural network architecture for learning human-object interactions," *Neurocomputing*, vol. 307, pp. 14–24, 2018.

[20] J. K. Moore, S. K. Hnat, and A. J. van den Bogert, "An elaborate data set on human gait and the effect of mechanical perturbations," *PeerJ*, vol. 3, p. e918, 2015.

[21] E. Diaconescu, "The use of NARX neural networks to predict chaotic time series," *WAEAS Transactions on Computer Research*, vol. 3, no. 3, pp. 182–191, 2008.

[22] O. Nerrand, P. Roussel-Ragot, D. Urbani, L. Personnaz, and G. Dreyfus, "Training recurrent neural networks: Why and How? An illustration in dynamical process modeling," *IEEE Transactions on Neural Networks*, vol. 5, pp. 178–184, 1994.

[23] M. Han, J. Xi, S. Xu, and F.-L. Yin, "Prediction of chaotic time series based on the recurrent predictor neural network," *IEEE Transactions on Signal Processing*, vol. 52, no. 12, pp. 3409–3416, 2004.

[24] X. Liao, G. Chen, and E. N. Sanchez, "LMI-based approach for asymptotically stability analysis of delayed neural networks," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 49, no. 7, pp. 1033–1039, 2002.

[25] H. P. H. Anh and N. Phuc, "Inverse neural MIMO NARX model identification of nonlinear system optimized with PSO," in 5^{th} *International Symposium on Electronic Design, Test & Applications*, 2010, pp. 144–149.

[26] S. Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd ed. Prentice Hall, 1998.

[27] W. L. Wu and F. C. Su, "Potential of the back propagation neural network in the assessment of gait patterns in ankle arthrodesis," *Clinical Biomechanics*, vol. 15, no. 2, pp. 143–145, 2000.

[28] M. Louzazni, H. Mosalam, A. Khouya, and K. Amechnoue, "A non-linear auto-regressive exogenous method to forecast the photovoltaic power output," *Sustainable Energy Technologies and Assessments*, vol. 38, p. 100670, 2020.

[29] T. Lin, B. Horne, P. Tino, and C. Giles, "Learning long-term dependencies in NARX recurrent neural networks," *IEEE Transactions on Neural Networks*, vol. 7, no. 6, pp. 1329–1338, 1996.

[30] S. C. Prasad and P. Prasad, "Deep recurrent neural networks for time series prediction," *arXiv preprint arXiv:1407.5949*, 2014.

[31] P. Baldi and A. F. Atiya, "How delays affect neural dynamics and learning," *IEEE Transactions on Neural Networks*, vol. 5, no. 4, pp. 612–621, 1994.

[32] T. Iwai, Y. Ohno, A. Niwa, Y. Nakamura, K. Sakai, K. Matsui, and H. Nishi, "Self-organizing map using classification method for services in multilayer computing environments," in *IECON 2018-44th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 2018, pp. 4193–4198.

[33] T. M. Nam, P. H. Phong, T. D. Khoa, T. T. Huong, P. N. Nam, N. H. Thanh, L. X. Thang, P. A. Tuan, V. D. Loi *et al.*, "Self-organizing map-based approaches in ddos flooding detection using sdn," in *2018 International Conference on Information Networking (ICOIN)*. IEEE, 2018, pp. 249–254.

[34] J. Vesanto, "Som-based data visualization methods," *Intelligent data analysis*, vol. 3, no. 2, pp. 111–126, 1999.

[35] M. Beale, M. Hagan, and H. Demuth, "Deep learning toolbox users guide," *The Mathworks Inc.: Herborn, MA, USA*, 2020.

[36] H. Mohammadi and H. Richter, "Robust tracking/impedance control: Application to prosthetics," in *American Control Conference*, July 2015, pp. 2673–2678.