SEMARAK ILMU PUBLISHING

Journal of Advanced Research in Applied Sciences and Engineering Technology

Journal homepage: https://semarakilmu.com.my/journals/index.php/applied_sciences_eng_tech/index
ISSN: 2462-1943

Predictive performance of MLP, Random Forest, CNN on knee joint gait angles with single IMU data

Nezar Elghazy[1,*], Ahmed Elsawaf[1], Moustafa A. Fouz[1]

[1] Mechanical Engineering Department, Arab Academy for Science, Technology and Maritime Transport, Cairo, Egypt

ARTICLE INFO

Article history:
Received 24 April 2026
Received in revised form 18 May 2026
Accepted 30 May 2026
Available online 16 June 2026

Keywords:
Machine learning, gait, trajectory prediction

ABSTRACT

Lower limb rehabilitation using active devices faces numerous challenges due to the complexities and uncertainties in human walking patterns. Accurate prediction of gait parameters with minimal subsystems installed in the device is essential. Recent research has focused on gait determination, including locomotion classification, intention detection, and human joint trajectory prediction, which is the focus of this study. This research compares three machine learning algorithms—MLP, CNN, and random forest—for predicting future knee gait angles using data from a single IMU sensor. The MLP model outperformed the other two, achieving the lowest RMSE of 0.0109, compared to 0.0174 for the random forest model and 0.0501 for the CNN. Additionally, the MLP model had a lower execution time than the CNN model. Although the random forest model had the fastest execution time, its large model size was impractical.

1. Introduction

Lower limb rehabilitation is a very relevant topic for the clinical field. It deals with patients with motor dysfunction, which could be caused by several common factors such as spinal cord injuries or strokes [1,2]. Exoskeletons for rehabilitation systems mainly aim to either help the patient regain their motor abilities or to constantly provide partial assistance to individuals to prevent muscle decay over time [2,3]. However, this has proven to be a rather complicated task since it involves interaction between the robot, the human impaired individual, and the environment, which is a task filled with many uncertain parameters. Therefore, recently, many researchers have explored the impact of AI algorithms to aid with such tasks as it has been shown in different engineering fields [4,5]. In summary, AI helps in one of four ways: Locomotion Classification (LC), Intention Detection (ID), Robot Control (RC), and Human Joint Trajectory Prediction (HJTP) [6]. To achieve these, machine learning models such as Reinforcement Learning and Neural Networks have been experimented with the most [6]. Training machine learning models requires datasets that, for example, refer to the gait cycle of

* *Corresponding author.*
E-mail address: Eng_nezargazy@aast.edu

https://doi.org/10.37934/araset.60.5.282291

human subjects. The data acquired was mostly gyroscopic readings, inertial readings, and EMG readings [7-12]. The focus of this research is HJTP.

Mohamed *et al* [13]. developed MLP and CNN models that intake IMU readings to predict future gait parameters on foot and compared the results to those of an LSTM neural network. The models were tested on a microcontroller, and the runtime of each model was measured, which benefits system compactness. Both models achieved a 0.98 correlation coefficient when the current gait phase was included, compared to 0.973 and 0.945 for CNN and MLP, respectively, and achieved 2.4 ms and 142 ms execution time on the microcontroller.

Wang *et al* [7]. proposed a data augmentation algorithm that is GANs-based (Generative Adversarial Networks). This was developed to enhance data acquisition usability since it is difficult and costly to conduct gait dataset acquisition. The experiments were done on 5 healthy individuals, and the data was extracted via an infrared optical motion capture system retrieving the position coordinates of each of the 6 markers placed on the human legs (3 per leg). The LSTM neural network was fed both raw data and augmented data, and the future gait trajectory and prediction metrics were compared.

Bian *et al* [14]. constructed a method for future joint angle prediction using only kinematic values. Data acquisition took place from IMU sensor readings and an optical motion detection system. The future angles were predicted by the machine learning model that consists of a stacked LSTM neural network. The prediction time is 10 ms ahead with an average mean square of 5.3 degrees and a coefficient of determination of 0.81.

This study aims to construct and compare machine learning algorithms for future knee gait angle prediction using the least number of mounted sensors. During this research, only one IMU mounted on the subject's thigh is utilized to capture the motion dynamics fed to the AI models. MLP, CNN, and random forest models are chosen for training on the selected dataset, and their performance is compared against each other.

2. Methodology

2.1 Utilized dataset

For this research the dataset developed by Santos *et al* [15] was used to implement future gait angle prediction. The experiments were conducted on 25 healthy individuals aged 18 to 47 years. Each participant had a band attached containing an IMU and a microcontroller. The IMU was used to collect the inertial and gyroscopic data from the human leg's thigh. Additionally, an optical motion detection system was utilized with 47 reflective markers fixed on predetermined positions on the participants' bodies with consistency for all individuals, as indicated in **Error! Reference source not found.**(a) [15]. Five of the reflective markers were placed on the band that holds the IMU in position. The optical system records the three-dimensional trajectory for each reflective marker. The trials were conducted so that each participant walked at their normal walking speed in a straight line for 5 meters. The marker position trajectories and IMU readings were synchronized and concatenated by frames. The sampling frequency for the recording was set to 100Hz.

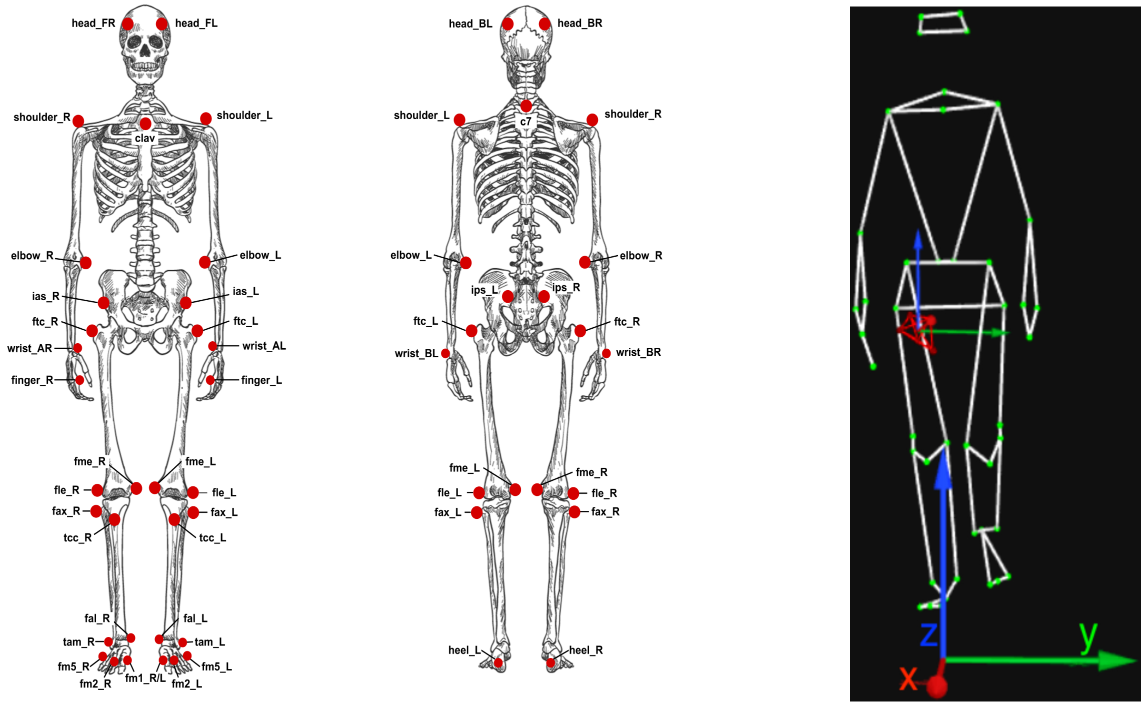
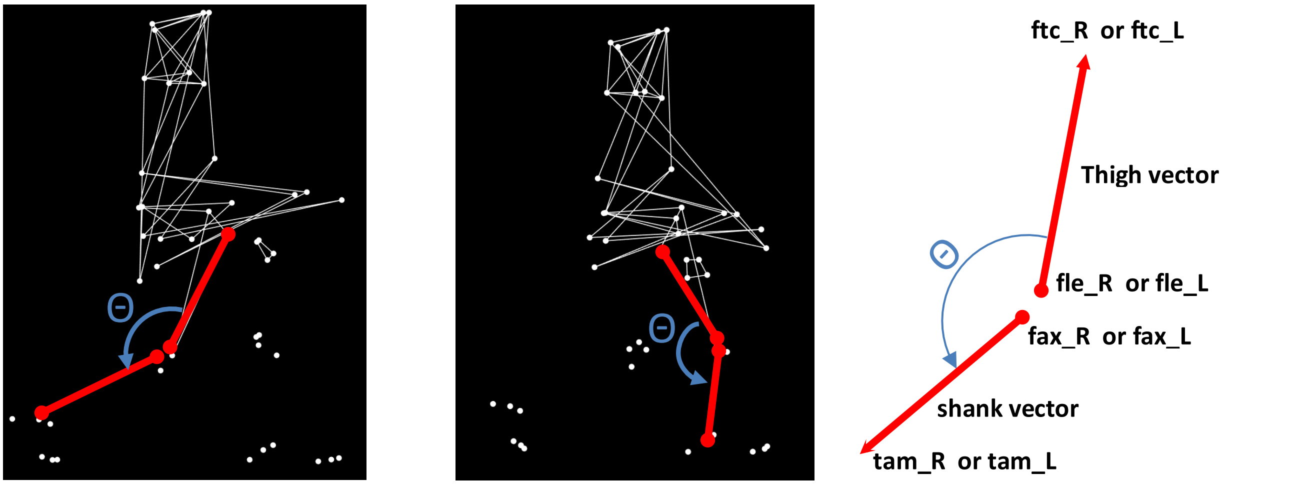


Fig.1 (a) Reflective markers positions [15], (b) World frame and band frames [15]

The optical data was labelled and filtered, then imported into MATLAB [16] to be available for retrieval and analysis using MoCap Toolbox [17]. **Error! Reference source not found.**(b) illustrates the world frame, where the X-axis is the walking direction. The IMU values were acquired by the microcontroller ESP8266, and the data was then saved in CSV files on a computer. Each subject performed 5 trials for each leg side per day for 2 days, totalling 20 samples.

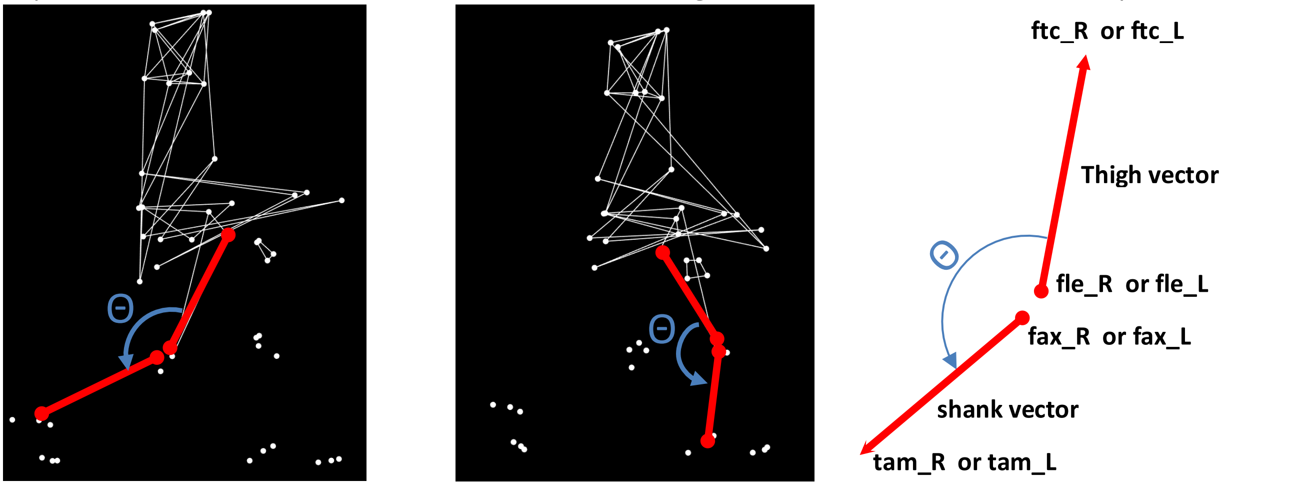
2.2 Dataset Preprocessing
2.2.1 Calculating knee angle

The prediction of future knee angles is essential for later control purposes, which is the angle between the thigh and shank. The leg side on which the band containing the IMU was installed is determined by comparing the reflective markers' y-axis coordinates on the band with the reflective markers installed on the left and right legs.



(a) (b) (c)

Fig. 2(a, b) shows a sample of the markers plotted in two different frames from one of the user's experiments. The chosen markers of the thigh and shank lines are represented in



(a) (b) (c)

Fig. 2(c). The angle between the thigh and the knee for each leg can then be calculated by performing the dot product between the presented two vectors.

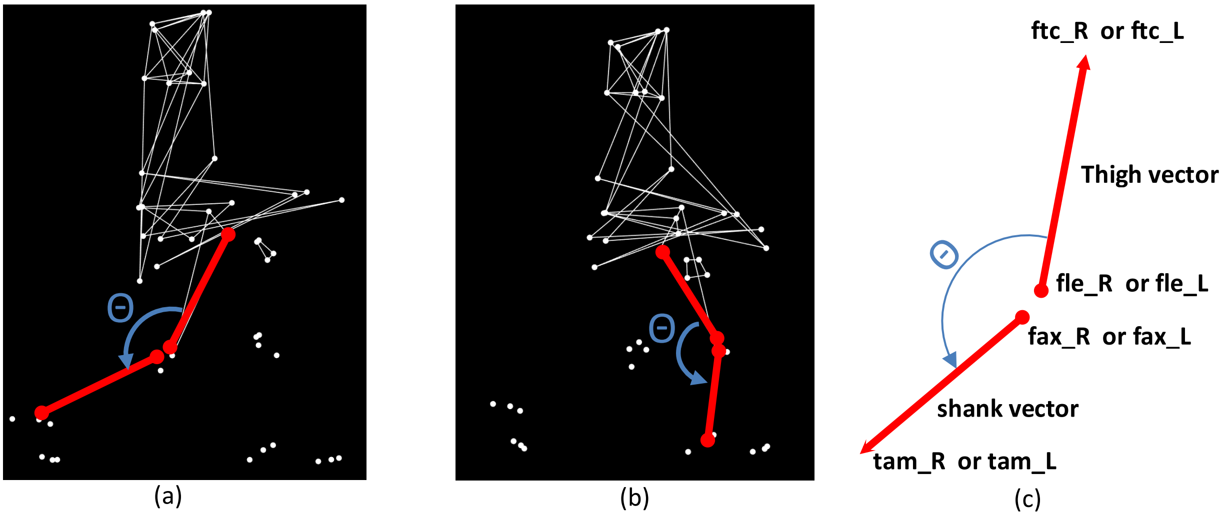


Fig. 2 (a) User01 Day 1, 200th frame's markers plot from MoCap, Matlab (b) User01 Day 1, 250th frame's markers plot from MoCap, Matlab MoCap, Matlab (c) Thigh and shank vectors with markers designated labels

$$\Theta_{knee} = cos^{-1}\left(\frac{\overrightarrow{thigh}\cdot\overrightarrow{shank}}{\|thigh\| \times \|shank\|}\right) \tag{1}$$

2.2.2 Preparing the dataset

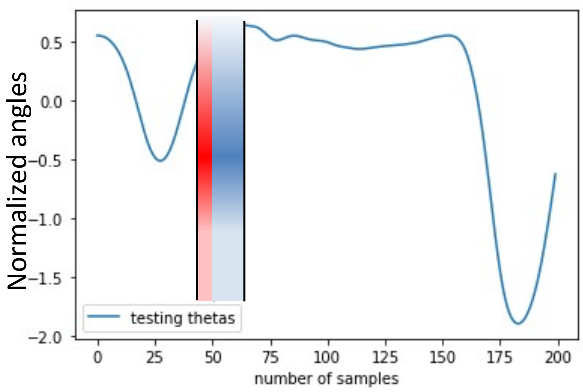
A moving average filter [18] was used to smooth out the IMU readings to counter the inertial drift, the equation used is.

$$z(n) = \frac{1}{p+1}\sum_{j=0}^{p} x(n-j) \tag{2}$$

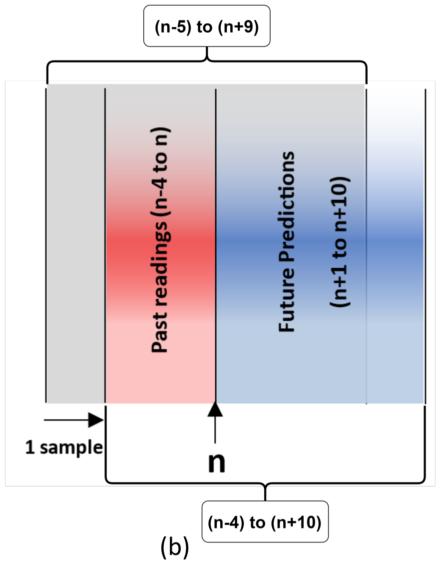
Where, P is window length, x is the IMU values, n is the sample number, and $0 \leq j \leq P$. The data was then normalized using z-score normalization so that the mean is zero and the standard deviation is one.

2.3 Machine Learning Models Construction

As aforementioned, the purpose of the machine learning models is to predict future gait knee angles to compensate for control delay. **Fig.3**(a) illustrates normalized knee angles with sample number, which was calculated from the markers as elaborated before. The input window (in red) to the model is the past five samples including the current values. Where, each sample includes all six IMU readings (three accelerometer readings and three gyroscope readings). The model output predicted values (in blue) would be the consequent ten future knee angle samples. The input and output windows are shifted one time step forward for each iteration, which can be designated as a moving window with respect to time as illustrated in **Fig.3**(b). The dataset was designed to feed the model in a moving window manner to emulate the operation of the control loop during the real-life process.



(a)



(b)

Fig.3 (a) Past and future window of the gait angle feature, (b) Training features and labels' incremental shift

2.3.1 Multi-layered perceptron model (MLP)

MLP is the most commonly used neural network model for knee rehabilitation purposes such as gait classification, parameter prediction, and control. This is due to its non-complexity compared to other deep learning models, yet it yields good results [6]. An MLP model is a network that connects input and output parameters, describing the relation between them with nodes known as neurons and weights connecting the nodes. Two main operations are involved in training the model: forward calculation of output based on input data, and backward propagation, which updates the weights values based on the predetermined cost function [19]. Fig.4 illustrates the developed MLP construction. The input to the model is 35 values representing five samples, where each sample is composed of six IMU readings and the corresponding calculated knee angle. The model has four hidden layers, two batch normalization layers, and one dropout layer. The activation function for all hidden layers and the output layer is a linear activation function, which was chosen since it preserves the negative values that comprise many of the dataset's values. The cost function is mean absolute error.

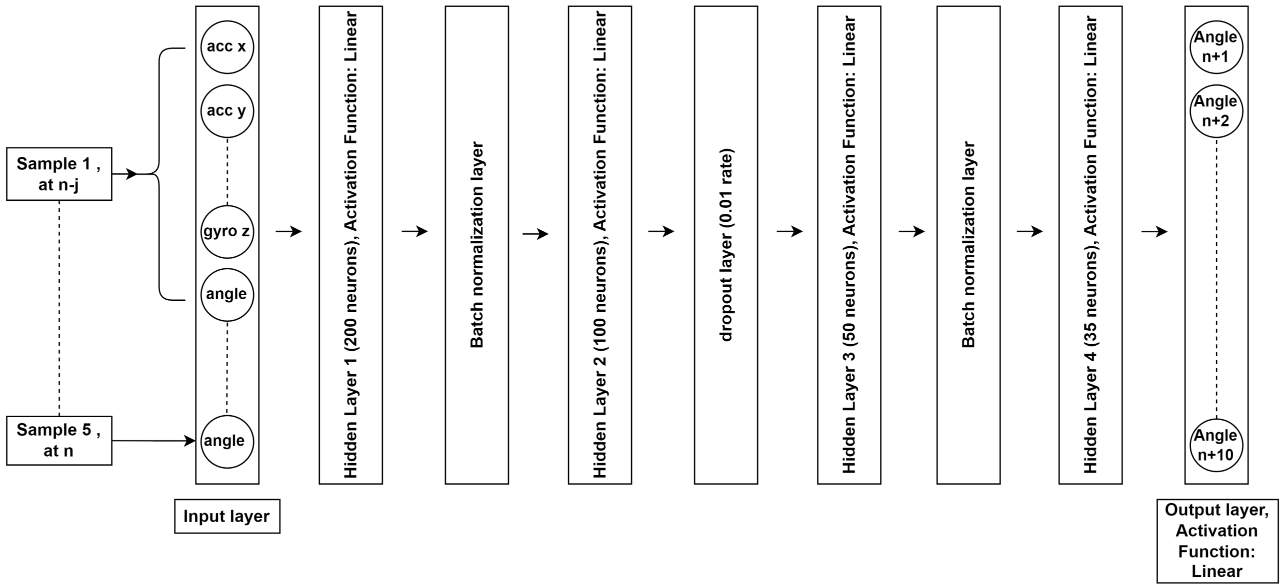


Fig.4 chosen MLP construction diagram

2.3.2 Random forest

A decision tree is another algorithm used in classification as well as regression tasks [6]. One implementation of the decision tree algorithm is the random forest model, which overcomes decision tree limitations such as overfitting. It is built by bootstrapping the dataset, which selects samples for training randomly. Decision trees are created using the bootstrapped data and random variables. This process is repeated for a specified number of trees, and the results are decided by voting among the created trees. For this research, a random forest model consisting of 100 decision trees was created.

2.3.3 Convolution neural network (CNN)

CNN is another model used in the field of gait prediction and lower limb exoskeleton system control in general [7]. CNN is a more complex machine learning model that utilizes convolution layers to extract features from the data by training, as well as including fully connected layers like in MLP [20]. The shifting window input to the CNN model and its output are the same as the MLP model. The construction of the CNN model is provided in Fig. 5. The model consists of 3 convolution layers, 2 max pooling layers, 2 dropout layers, 1 flatten layer, 2 fully connected dense layers, and the output layer.

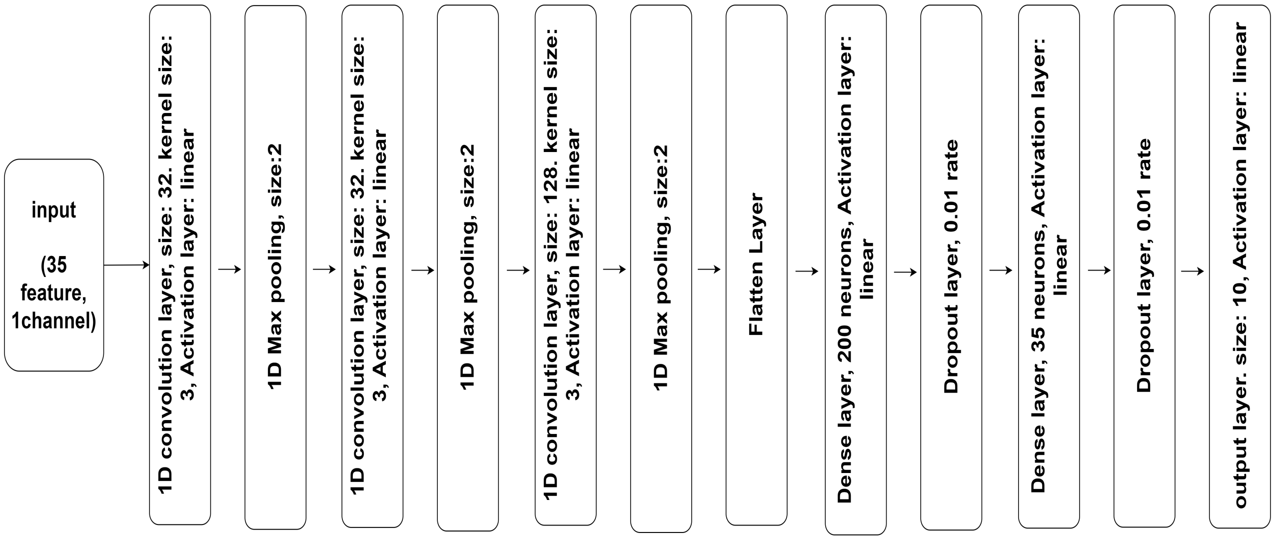


Fig. 5 chosen CNN construction diagram

2.3.4 Utilized evaluation methods

Since these models predict continuous values and this isn't a classification problem, the chosen evaluation metrics are [21,22]:

- Mean absolute error (MAE)

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|\Theta_{acual} - \Theta_{predicted}\right| \quad (3)$$

MAE measures the average of residuals in the dataset, which is the average of the absolute difference between actual angles and predicted angles.

- Root mean squared error (RMSE)

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(\Theta_{i,acual} - \Theta_{i,predicted}\right)^2} \quad (4)$$

RMSE measures the square root of the mean squared error, which measures the standard deviation of the residuals.

- Coefficient of determination (R^2)

$$R^2 = 1 - \frac{\sum_{i-1}^{n}\left(\Theta_{acual} - \Theta_{predicted}\right)^2}{\sum_{i-1}^{n}(\Theta_{acual} - \overline{\Theta_{acual}})^2} \quad (5)$$

The coefficient of determination explains how well the predicted values explain the variation in the actual values. A value of $R^2 = 0$ means the model always predicts the mean value, while a value of $R^2 = 1$ means the model perfectly predicts the variation in the data.

- Pearson correlation coefficient (CC)

$$CC = 1 - \frac{cov(\Theta_{acual}, \Theta_{predicted})}{\sigma_{\Theta_{acual}} \sigma_{\Theta_{predicted}}} \tag{6}$$

The Pearson correlation coefficient is a measure of how two variables form a linear relationship. It is computed by calculating the ratio between the covariance of the actual and predicted angles and the product of their standard deviations.

3. Results

The models were run using Python 3.7 on a workstation (GPU RTX 3060 TI, 48 GB RAM, CPU intel i5-10400F). The code was run with Spyder IDE in Anaconda. For the MLP and CNN, the training took place over 1000 epoch. Table 1 shows the comparison between the performance of the different studied models.

Table 1
Evaluation metrics, MLP vs Random Forest

Metric	MLP	Random Forest	CNN
MAE	0.0088	0.011	0.0374
RMSE	0.0109	0.0174	0.0501
R^2	0.99	0.97	0.99
CC	0.98	0.98	0.99
File size	170 KB	2.4 GB	385 KB
Execution time	0.135 seconds	0.006 seconds	0.1579 seconds

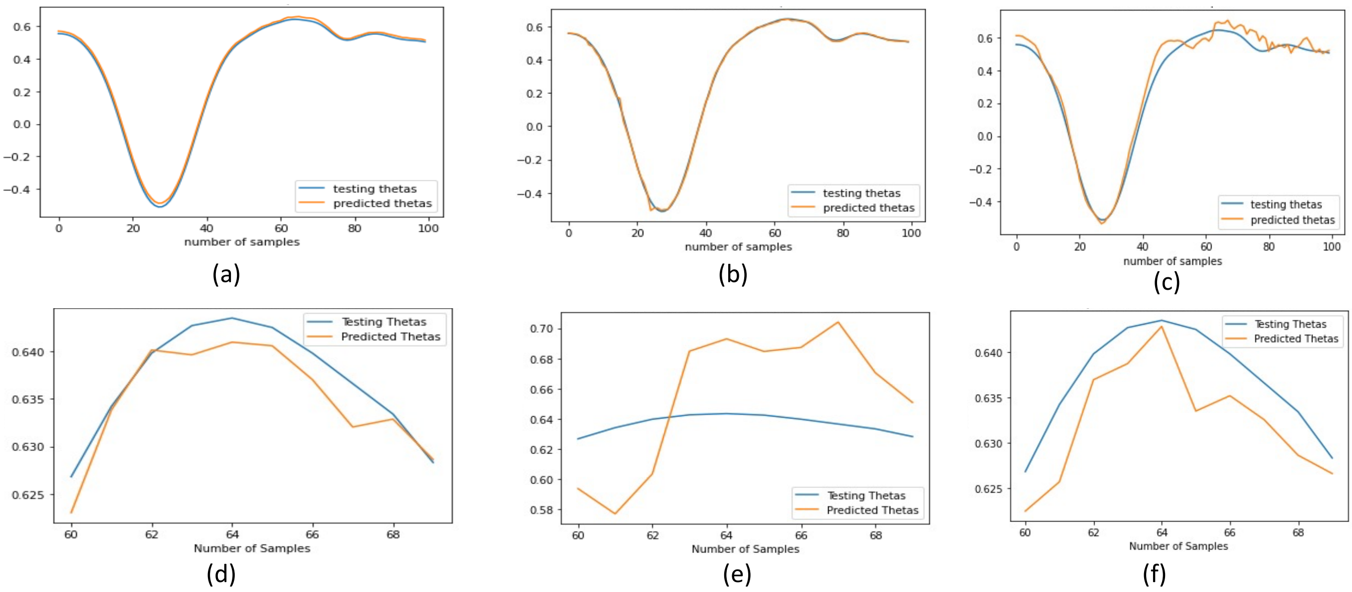


Fig.6 (a, d) MLP prediction against testing angles for 100 samples and 10 samples, (b, e) random forest prediction against testing angles for 100 samples and 10 samples, (c, f) CNN prediction against testing angles for 100 samples and 10 samples respectively

4. Conclusions

The results show that the MLP model was superior to Random Forest and CNN in terms of accuracy; it had the least error and the highest coefficient of determination and correlation coefficient. Additionally, the MLP model had a quicker execution time and a smaller model size on file compared to the CNN. The Random Forest model's execution time was much quicker than both the MLP and CNN models. However, since the tree size corresponds to the dataset size, the file size became large, which is not ideal for practical use.

Acknowledgement
This research was not funded by any grant

References
[1] Zhou, Jinman, Shuo Yang, and Qiang Xue. 2021. "Lower Limb Rehabilitation Exoskeleton Robot: A Review." *Advances in Mechanical Engineering* 13 (4): 1–17. https://doi.org/10.1177/16878140211011862.

[2] Shi, Di, Wuxiang Zhang, Wei Zhang, and Xilun Ding. 2019. "A Review on Lower Limb Rehabilitation Exoskeleton Robots." *Chinese Journal of Mechanical Engineering* 32 (1). https://doi.org/10.1186/s10033-019-0389-8.

[3] Zhang, Ting, Minh Tran, and He Huang. 2018. "Design and Experimental Verification of Hip Exoskeleton with Balance Capacities for Walking Assistance." *IEEE/ASME Transactions on Mechatronics* 23 (1): 274–85. https://doi.org/10.1109/TMECH.2018.2790358.

[4] Mohd Ariffin, Noor Afiza, Usman Abdul Gimba, and Ahmad Musa. 2025. "Face Detection Based on Haar Cascade and Convolution Neural Network (CNN)". *Journal of Advanced Research in Computing and Applications* 38 (1):1-11. https://doi.org/10.37934/arca.38.1.111

[5] Shatnawi, Hashem, and Mohammad N. Alqahtani. 2025. "Delving into the Revolutionary Impact of Artificial Intelligence on Mechanical Systems: A Review". *Semarak International Journal of Machine Learning* 1 (1):31-40. https://doi.org/10.37934/sijml.1.1.3140a.

[6] Coser, Omar, Christian Tamantini, Paolo Soda, and Loredana Zollo. 2024. "AI-Based Methodologies for Exoskeleton-Assisted Rehabilitation of the Lower Limb: A Review." *Frontiers in Robotics and AI* 11 (February): 1–19. https://doi.org/10.3389/frobt.2024.1341580.

[7] Wang, Yan, Zhikang Li, Xin Wang, Hongnian Yu, Wudai Liao, and Damla Arifoglu. 2021. "Human Gait Data Augmentation and Trajectory Prediction for Lower-Limb Rehabilitation Robot Control Using GANS and Attention Mechanism." *Machines* 9 (12). https://doi.org/10.3390/machines9120367.

[8] Wang, Fangzheng, Lei Yan, and Jiang Xiao. 2019. "Human Gait Recognition System Based on Support Vector Machine Algorithm and Using Wearable Sensors." *Sensors and Materials* 31 (4): 1335–49. https://doi.org/10.18494/SAM.2019.2288.

[9] Yang, Yong, Deqing Huang, and Xiucheng Dong. 2019. "Enhanced Neural Network Control of Lower Limb Rehabilitation Exoskeleton by Add-on Repetitive Learning." *Neurocomputing* 323: 256–64. https://doi.org/10.1016/j.neucom.2018.09.085.

[10] Hasanzadeh Fereydooni, Rohollah, Hassan Siahkali, Heidar Ali Shayanfar, and Amir Houshang Mazinan. 2020. "SEMG-Based Variable Impedance Control of Lower-Limb Rehabilitation Robot Using Wavelet Neural Network and Model Reference Adaptive Control." *Industrial Robot* 47 (3): 349–58. https://doi.org/10.1108/IR-10-2019-0210.

[11] Ge, Wangyang, Juan Zhao, Feng Wang, Chi Xu, Zhaohui Yang, and Jinhua She. 2022. "Experimental Design of Lower-limb Movement Recognition Based on Support Vector Machine." In *Proceedings of the 2022 41st Chinese Control Conference (CCC)*, July. https://doi.org/10.23919/ccc55666.2022.9902297.

[12] Tang, Kunhao, Ruogu Luo, and Sanhua Zhang. 2021. "An Artificial Neural Network Algorithm for the Evaluation of Postoperative Rehabilitation of Patients." *Journal of Healthcare Engineering* 2021 (October): 1–6. https://doi.org/10.1155/2021/3959844.

[13] Karakish, Mohamed, Moustafa A. Fouz, and Ahmed ELsawaf. 2022. "Gait Trajectory Prediction on an Embedded Microcontroller Using Deep Learning." *Sensors* 22 (21). https://doi.org/10.3390/s22218441.

[14] Bian, Qingyao, Marco Castellani, Duncan Shepherd, Jinming Duan, and Ziyun Ding. 2024. "Gait Intention Prediction Using a Lower-Limb Musculoskeletal Model and Long Short-Term Memory Neural Networks." *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 32: 822–30. https://doi.org/10.1109/TNSRE.2024.3365201.

[15] Santos, Geise, Marcelo Wanderley, Tiago Tavares, and Anderson Rocha. 2022. "A Multi-Sensor Human Gait Dataset Captured through an Optical System and Inertial Measurement Units." *Scientific Data* 9 (1): 1–10. https://doi.org/10.1038/s41597-022-01638-2.

[16] MATLAB. n.d. *The MathWorks, Inc.*

[17] Toiviainen, Petri. 2024. *Mocaptoolbox*. Accessed July 10, 2024. https://github.com/mocaptoolbox.

[18] Lynch, Kevin M., Nicholas D. Marchuk, and Matthew L. Elwin. 2015. *Embedded Computing and Mechatronics with the PIC32 Microcontroller*. https://openlibrary.org/books/OL28531141M/Embedded_Computing_and_Mechatronics_with_the_PIC32_Microcontroller.

[19] Marsland, Stephen. 2009. *Machine Learning: An Algorithmic Perspective*. https://www.comp.hkbu.edu.hk/~cib/2009/Dec/review1/iib_vol10no1_review1.pdf.

[20] Ketkar, Nikhil. 2017. *Deep Learning with Python: A Hands-on Introduction*. Berkeley, CA: Apress.

[21] Dey, Sharmita, and Arndt F. Schilling. 2023. "A Function Approximator Model for Robust Online Foot Angle Trajectory Prediction Using a Single IMU Sensor: Implication for Controlling Active Prosthetic Feet." *IEEE Transactions on Industrial Informatics* 19 (2): 1467–75. https://doi.org/10.1109/TII.2022.3158935.

[22] Su, Binbin, and Elena M. Gutierrez-Farewik. 2020. "Gait Trajectory and Gait Phase Prediction Based on an LSTM Network." *Sensors* 20 (24): 1–17. https://doi.org/10.3390/s20247127.