POLITECNICO MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

A Kinematic-based Intention Detection Strategy for Robotic Assistance: A Real-Time Approach to Upper Limb Rehabilitation

Laurea Magistrale in Biomedical Engineering - Ingegneria Biomedica

Author: Luca Cagliani,987533 Alessandro Scarpelli, 990272

Advisor: Prof. Emilia Ambrosini

Co-advisor: Federica Ferrari

Academic year: 2022-2023

1. Introduction

Neurological disorders, like stroke, result more present in the actual society due to the aging of global populations, sedentary and the current lifestyle. Furthermore, thanks to the improvement of medical care, more people affected by these conditions can survive, thus innovative techniques in order to promote the motor recovery of the functionality of the affected limbs, are essential. [1] Withing this frame, one innovative technology which aims to promote motor recovery of people affected by these types of disorders is rehabilitation robotics, regarded as the application of robotics devices in motor rehabilitation therapies. Rehabilitation robotics does not want to replace the figure of therapist, however being a valid support of the therapy. Indeed, exoskeletons show important advantages in the rehabilitation process like high dosage and the repeatability of exercise in safe conditions and the customization of the training. The key aspect of the rehabilitation process is to exploit the active participation of the patient, indeed, according to literature, the probability of the motor recovery of the subject with an active participation results much higher. [7] Finally linking the advantages of the exoskeleton and the so-called "intention detection strategies" it is possible to provide the awareness and the dignity of executing daily life activities to impaired people.

1.1. Intention detection strategies

Intention detection strategies are methods or approaches used to recognize the volitional movements or intentions of an individual. It serves as the bridge between human intention and technological response. According to the literature, different types of intention detection strategies can be performed: [4]

- Bio-Signal-Based IDS: Detection of signals like EEG and EMG for direct device control, applicable in mind-controlled prosthetics and brain-computer interfaces. However, EMG signals have limitations, such as sensitivity to electrode placement and muscle fatigue.

- Vision-Based IDS: Camera systems with artificial vision detect body movements, used in gesture recognition and motion tracking, including video games. Eye-tracking techniques, like video-oculography and electrooculography, aid motion planning, benefiting those with limited upper limb functionality due to neurological deficits.

- Inertial Sensor-Based IDS: Inertial sensors, found in wearables like smartwatches and smartphones, use gyroscopes and accelerometers to monitor body orientation and acceleration. Widely applied in rehabilitation for motion monitoring, they are easy to wear but may be sensitive to electronic interference, and require calibration.

- Voice-Based IDS: This IDS uses vocal sounds and voice commands, ideal for hands-free control for individuals with speech abilities. The number of distinguishable states is practically limited by computational power and software capabilities, with potential accuracy issues in noisy environments.

- Upper Limb Movement IDS: This approach detects intentional movement through joint rotation, facilitating natural interaction between individuals and technology for those with limited upper limb function. However, it's more practical for users with some residual upper-limb function and may not be suitable for severe impairments or full paralysis.

1.2. Related works

The idea of this study was to exploit a kinematic and torque-based ID for upper limb reaching frontal rehabilitation exercises using machine learning models. The workflow begins with a literature review to identify the most accurate models and their latency in recognizing reaching tasks. Three studies are presented:

First study: [5]

-Objective: Evaluate real-time classification accuracy for upper-limb robotic prosthesis reaching tasks with ten healthy subjects.

-Methods: Used inertial sensors and a camera-based vision system, testing various object positions and orientations.

-Results: KNN and Random Forest were the best classifiers, with KNN achieving nearly 90% accuracy.

Second study: [6]

-Objective: Classify upper extremity exercises using IMU-based kinematic data.

-Methods: Involved fifty healthy participants performing various arm exercises (bicep curl, frontal arm raise ecc..) with IMUs capturing joint angles.

-Results: KNN and Random Forest were the most accurate classifiers, approximately 92% for upper limb exercises.

Third study: [8]

-Objective: Evaluate machine learning models for real-time human activity recognition using raw IMU data.

-Methods: Tested activities like standing, running, and walking with two healthy participants wearing IMUs on their chest, thigh, and tibia.

-Results: SVM with radial kernel achieved the best accuracy (97%) for recognizing human activities, with an average latency of approximately 107.1 ms.

In conclusion, across these studies, KNN and Random Forest showed the best accuracies for reaching task of upper limb exercises. Additionally, the study on real-time activity recognition highlighted the importance of low latency, with SVM showing promising results in timely feedback applications.

1.3. Aim of the works

The goal of the study was to develop and validate an effective kinematic and torque-based intention detection strategy for upper limb rehabilitation using a robot-assisted exoskeleton for mild impairment patient. Three distinct tasks, starting from a resting position, were identified through the IDS to initiate specific upper limb movements during rehabilitation therapy. Ten healthy subjects wore the exoskeleton, and data collected from integrated sensors were analysed to develop and test the best machine learning model, minimizing the latency, including KNN, Random Forest, Gradient Boosting, Extremely Randomized Trees, Extreme Gradient Boosting, and Long Short-Term Memory. The chosen model was implemented in the robotic platform AGREE and evaluated in real-time with four healthy subjects. The study contributes to the literature by comparing results with existing state-of-the-art techniques.

2. Material and methods

2.1. Experimental setup

The "AGREE" (Arm exoskeleton and Grip assistance for REhabilitation and indipEndent living) is an advanced powered exoskeleton designed to assist patients with neurological conditions during seated rehabilitative exercises. These exercises include activities like hand-to-mouth movements, lateral elevation, and arm reaching. One notable feature of the AGREE device is its adaptability to work with both the right and left upper limbs, making it a versatile tool for rehabilitation for a wide range of patients with different needs and conditions. The primary goal of the AGREE exoskeleton is to provide advanced support for patients during rehabilitation exercises, ultimately helping them enhance their mobility and achieve greater independence in their daily activities. The AGREE device boasts four degrees of freedom, which include three actuated joints at the shoulder and one actuated joint at the elbow, with an additional passive joint for the forearm. The mechanical design is centered around four active joints: [3]

- J1: shoulder horizontal abduction/adduction.
- J2: shoulder flexion/extension.
- J3: intra/extra humeral rotation abduction.
- J4: elbow flexion/extension.

Notably, the shoulder's flexion/extension joint features a passive anti-gravity system that utilizes springs to counteract gravitational forces acting on the arm. The exoskeleton's design closely mirrors the natural human joint structure and provides support for the user's arm at two interface points: one on the upper arm and the other on the forearm. It also has a sampling frequency of 1000 Hz. Angular position, rotational velocity, and output torque are gathered for each joint of the exoskeleton. Incremental encoders, specifically MILE encoders from Maxon Motor in Switzerland, record position and velocity. Torsional load cells, specifically TRT load cells from Transducer Techniques in the United States, acquire output torque at the actuator's output shaft. Subtracting the gravitational torque caused by the robot's weight from the joint load cell torque provides a measure of the human-robot interaction effort. This design ensures that the exoskeleton can adapt compliantly to forces generated by the user and those applied by the therapist. Furthermore, mechanical end-stops have been incorporated into each joint to prevent the arm from moving beyond its natural range of motion, emphasizing safety during rehabilitation exercises.

2.2. Experimental protocol

Thirteen healthy subjects (age: 24 ± 0.94, 8 male, 5 female, right hand) were involved in the testing protocol. Nine subjects were used for data collection, while four were involved in the real-time evaluation with AGREE. The protocol was performed with the explicit consent of the participants and it was approved by the Ethical Committee of Politecnico di Milano. Each subject wore the exoskeleton on their right side, with the rotational axes of the anatomical joints aligned with the robot's ones. The protocol consisted of performing 3 reaching tasks towards three different target positions: contralateral to the right limb, frontal and ipsilateral. Each movement was composed as follows: the subject started from the rest position, where the arm was placed in front of the body to maintain the elbow joint at almost 90° of flexion, reached the target point and returned to the rest position to rest for at least 3s. Each movement were repeated 15 times for each target position. The movements were executed with the exoskeleton with null arm weight compensation and in transparent mode, therefore the subject did not have to follow a predefined trajectory but could choose the height and speed for the completion of the task. The dataset created contained all the information concerning the kinematics and torques of each AGREE joint. The workflow that was followed began with a processing of the data in Matlab with the aim of extracting windows corresponding to the start of the movement and the rest position for all executed movements. Joint J3 was keeping locked during data acquisition phase thus it was not considered during the data processing. Subsequently, this data was imported into Python to allow the training of machine learning models in the classification of the direction of the performed movement. Once the best model was found, it was tested in real-time on AGREE. Finally, the results obtained were compared with the state of the art.

2.3. Data Processing

Once the data were loaded into Matlab, the velocities of the joints of the exoskeleton were analysed. By applying a threshold of 0.05 rad/s, all points where the velocities of these joints were simultaneously zero were identified. In this process, a vector was generated containing a series of indices indicating the moments when the subject remained motionless. These included the 'rest positions', detected during longer intervals, and the moments when the subject reached the target point, during shorter intervals. After an initial selection, in which only the last value of each interval was extracted by identifying time jumps in the vector, several points were identified that generated sub-intervals. This occurred because the subject, in the two positions mentioned, could not remain completely still, generating velocities greater than 0.05 rad/s. So an algorithm was used, based on a 'for' loop, which had the following logic:

$$\begin{array}{c} \text{IF } (x+1) - x \leq \text{T AND } x \neq 0 \\ \text{THEN } x = 0 \end{array} \tag{1}$$

Where $x+1$ and x are two consecutive indices and T indicates the threshold, different for each subject and that in time had values around 2 seconds. The choice of threshold was determined through a graphical analysis of the differences between the points, choosing an intermediate value to distinguish the differences to be eliminated from those to be maintained. Through this approach, all indices representing the last point of each interval were identified. In the context of this study, it was not relevant to distinguish between the outgoing and return in the individual repetitions, so the indices that identified the attainment of the target point were eliminated. These steps made it possible to identify the instants at which the subject began each individual repetition for each direction of the reaching task.

From these points, different size of time windows has been created: 10ms, 20ms, 30ms, 50ms, 100ms, 150ms, 200ms and 300ms. The flow followed for their creation was as follows: for each repetition, the point identifying the instant at which the subject began the movement was taken, and from that point a window was extracted, containing a quantity of values allowing the identification of the beginning of the movement by the automatic classifiers and therefore each point in the window has been associated with the same class. Furthermore, to allow the models to recognize the rest position, the same window was extracted to the left of the instant at which the movement began to contain all points at which the velocities of the joints were zero and in this case each value was associated with the 'REST' class. Figure 1 shows the starting instant of movement in red and around the windows described above. A dataset was generated for each window created and for each subject. It contained the nine most relevant columns for describing the movement of the exoskeleton joints: the position, speed and torque for each joint considered and a column, the label, describing the class to which it belongs.

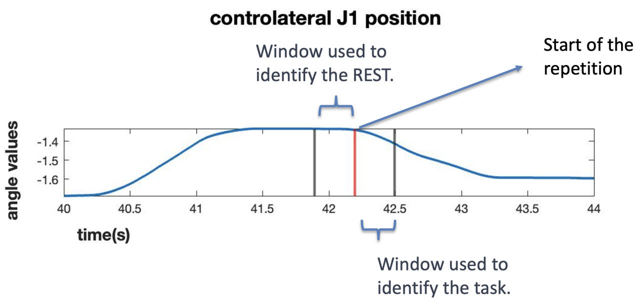

Figure 1: In the figure, the red vertical lines identify the points determined as the beginning of the movement, while the black ones represent the windowing considered for each repetition.

2.4. Model training

Once the datasets were loaded into Python, the subjects were divided as follows: eight were used as training sets, one as a test set and four to evaluate real-time performance. The models considered to classify the direction of movement performed by the subjects are: the K-Nearest Neighbors (KNN) model, the Random Forest (RF) model, Gradient Boosting (GB), Extremely Randomized Trees (Extra Trees), Extreme Gradient Boosting (Xgboost) and Long Short-Term Memory (LSTM). A grid search was used to identify optimal parameters for each model and accuracy and confusion matrix were used as evaluation metrics. The models that were tested were chosen from those most frequently used in the literature, like [2, 8–10].

2.5. Online testing

Once the results of the models listed above were compared, the best one was integrated on AGREE to perform the real-time evaluation. it is important to note that the model performs a pointwise recognition. This means that for

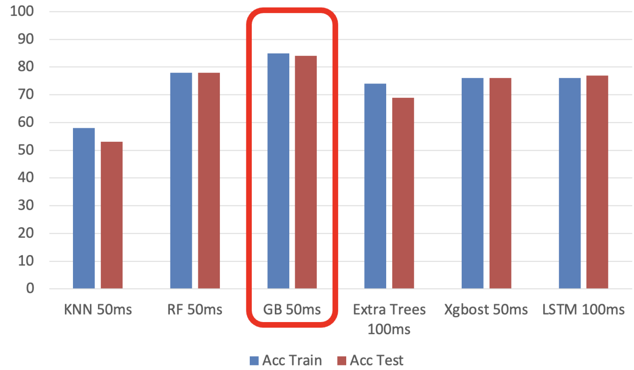

Figure 2: The figure shows the best result for each model with the respective window selected. The model circled in red, Gradient Boosting, was chosen for real-time evaluation and subsequent comparison in the literature.

each set of velocity, torque and position values of each joint it will make a prediction. To make the result more robust, logic was used such that the model had to perform a series of consecutive predictions of the same class to effectively classify the direction of movement made by the subject. In order to choose the number of consecutive predictions to be performed, different quantities of samples were tested using one subject: 5, 10, 20, 30, 40 and 50. They were evaluated according to the resulting accuracy. The result showed that the accuracy remained the same for the number of samples ranging from 5 to 30 and decreased in 40 and 50. With the same results, 5 was chosen as the number of samples used to predict the class as the speed of direction recognition is a key point in real-time applications. Thus, in addition to evaluating the performance of the movement direction recognition and REST phase by accuracy and confusion matrix, latency was also evaluated, calculated as the time in which the model identifies the task and communicates it to the exoskeleton.

3. Results and Discussion

3.1. Model results

The KNN presented a very low overall accuracy on all windows, train accuracy of less than 60%, and furthermore, overfitting increased with increasing window size. The one inherent in the 50ms window was chosen as the best result, with a test accuracy of 53%. The RF model showed a better overall accuracy, above 70% on both test and train, suffering less from overfitting than the previous model. The best result corresponds to the 50ms window with an accuracy on the test of 78%. The GB performs even better than the two previous models with no overfitting for all windows tested. The best results correspond to the 50ms, 100ms and 150ms windows. The 50ms window was chosen, with a test accuracy of 85%, because having less data allows faster model training. The Extra Trees, on the other hand, performed worse in terms of accuracy than the RF and GB with a test accuracy of no more than 70%. The 100ms window was chosen as the best result with an accuracy on the test of 69%. The Xgboost model showed good but worse performance than the normal GB presented above. The 50ms window was chosen as the best result with an accuracy on the test of almost 76%. Finally, the LSTM model presented a very low performance of 50% on the small windows, from 10ms to 30ms, and showed a very important presence of overfitting for the large windows, 200ms and 300ms. Therefore, the 100ms window was chosen as the best result with a test accuracy of almost 77%. The best results of the previously listed models were compared, and the best performing model is the GB with the 50ms window as shown in the Figure 2. The confusion matrix of the best performing model is shown in Figure 3.

3.2. Real-time evaluation

The GB model, which emerged as the best in the previous comparison, was integrated into AGREE for testing in real-time evaluation on 4 of the 13 subjects involved in the study. The results are displayed through the confusion matrix reported in Figure 4. It shows that the model

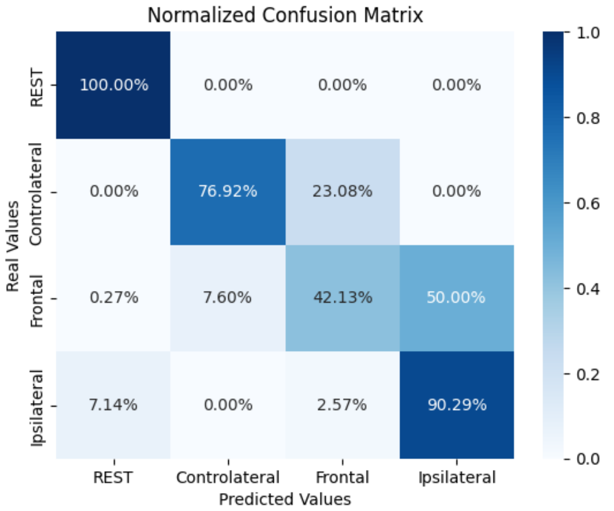

Figure 3: Confusion Matrix of Gradient Boosting, the best-performing model among those tested in the study.

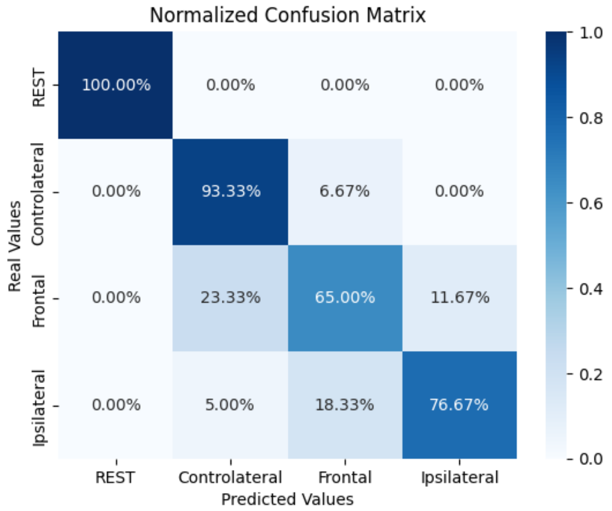

Figure 4: Confusion Matrix of Gradient Boosting tested in real-time evaluation in AGREE.

performs very well in classifying the REST position and contralateral direction. Regarding the frontal and ipsilateral directions, it can be observed that the performance is sub-optimal, which can be attributed to the lack of standardization in the positioning of the table during data acquisition, resulting in variations in the target point location. Although it remained in the frontal position, the exact location differed with each instance and as the classification of the direction is dependent on the movement of the shoulder, the latter may have been very similar in the two directions and worsened the performance of the model. On average, the time required for this process was 5.2ms. Thus, the overall accuracy associated with a latency of 5.2ms milliseconds was 84%, similar to that found in training.

4. Conclusions

The study aimed to assess the use of kinematics and torques parameters for recognizing frontal reaching movements through a robotic platform AGREE. Various machine learning models were tested, with the best result achieved using gradient boosting, which yielded an accuracy of 85%. However, the recognition of frontal movements had lower precision (approximately 42%) compared to other movements, likely due to variations in table positions and the execution of movements by healthy subjects. In a real-time evaluation four subjects were tested, the model achieved a final accuracy of 84% with a latency of 5,2 ms, similar compared to the offline evaluation. These results deviate from the literature, which typically uses EMG and IMU sensors, while this study solely relied on built-in exoskeleton sensors: infact from paragraph 1.2 the best accuracy achieved from literature is 92% while in this study was 84%. Regarding the latency, in this study an important result has been achieved with a final latency of 5,2 ms which is certainly lower than the one found in the literature (107,1 ms). In summary, the top-performing model in this study demonstrated a slightly lower accuracy compared to existing literature but excelled in rapidly recognizing intended movements, attributed to differences in data sources (kinematics vs. EMG, IMU, and built-in sensors) and a focus on low latency. For future development, enhancements could involve augmenting the dataset with more healthy subjects, aiming for greater heterogeneity to improve accuracy. Standardizing the table position for acquiring new subjects is proposed to enhance the accuracy of frontal movements, which exhibited lower performance among the three tasks. Additionally, evaluating the model on the AGREE robotic platform for the other two movement categories, hand-to-mouth and lateral elevation, is suggested.

References

[1] A Alawieh, J Zhao, and W Feng. Factors affecting post-stroke motor recovery: Implications on neurotherapy after brain injury. *Behav Brain Res*, 2016. Epub 2016 Aug 13.

[2] F Attal, S Mohammed, M Dedabrishvili, F Chamroukhi, L Oukhellou, and Y Amirat. Physical human activity recognition using wearable sensors. *Sensors*, 2015.

[3] S Dalla Gasperina et al. Agree: A compliant-controlled upper-limb exoskeleton for physical rehabilitation of neurological patients. *IEEE Transactions on Medical Robotics and Bionics*, 2023.

[4] J Gantenbein, J Dittli, JT Meyer, R Gassert, and O Lambercy. Intention detection strategies for robotic upper-limb orthoses: A scoping review considering usability, daily life application, and user evaluation. *Front Neurorobot*, 2022.

[5] M. Gardner, C.S. Mancero Castillo, S. Wilson, D. Farina, E. Burdet, B.C. Khoo, S.F. Atashzar, and R. Vaidyanathan. A multimodal intention detection sensor suite for shared autonomy of upper-limb robotic prostheses. *Sensors (Basel)*, Oct 2020.

[6] A. Hua, P. Chaudhari, N. Johnson, J. Quinton, B. Schatz, D. Buchner, and M.E. Hernandez. Evaluation of machine learning models for classifying upper extremity exercises using inertial measurement unit-based kinematic data. *IEEE J Biomed Health Inform*, 2020.

[7] L Marchal-Crespo and DJ Reinkensmeyer. Review of control strategies for robotic movement training after neurologic injury. *J Neuroeng Rehabil*, 2009. Jun.

[8] Q Mascret, M Bielmann, C L Fall, LJ Bouyer, and B Gosselin. Real-time human physical activity recognition with low latency prediction feedback using raw imu data. In *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2018.

[9] A Narayan, FA Reyes, M Ren, and Y Haoyong. Real-time hierarchical classification of time series data for locomotion mode detection. *IEEE J Biomed Health Inform*, 2022.

[10] C Pew and GK Klute. Turn intent detection for control of a lower limb prosthesis. *IEEE Trans Biomed Eng*, 2018.