IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 25 September 2023, accepted 18 October 2023, date of publication 30 October 2023,
date of current version 8 December 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3328254

RESEARCH ARTICLE

BiLSTM and SqueezeNet With Transfer Learning for EEG Motor Imagery Classification: Validation With Own Dataset

ALICIA GUADALUPE LAZCANO-HERRERA[1], RITA Q. FUENTES-AGUILAR[2], ADRIAN RAMIREZ-MORALES[3], AND MARIEL ALFARO-PONCE[2], (Member, IEEE)

[1]School of Engineering and Sciences, Tecnologico de Monterrey, Monterrey 64849, Mexico
[2]Institute of Advanced Materials for Sustainable Manufacturing, Tecnologico de Monterrey, Monterrey 64849, Mexico
[3]UPIITA, Instituto Politecnico Nacional, Mexico City 07340, Mexico

Corresponding author: Rita Q. Fuentes-Aguilar (rita.fuentes@tec.mx)

This work involved human subjects or animals in its research. Approval of all ethical and experimental procedures and protocols was granted by the "Comité Institucional de Ética en Investigación" de la Escuela de Ingeniería y Ciencias, del Instituto Tecnológico y de Estudios Superiores de Monterrey.

ABSTRACT Transfer Learning (TL) is a methodology that allows the re-train of a Machine Learning (ML) algorithm (like Neural Networks or NN's) for a new task with the advantage of the previous training acquired knowledge; with this methodology, it is possible to train NNs for a new task even if the data is scarce. The present study uses this approach to train NNs to classify Electroencephalography (EEG) signals that include Movement/Imagery (MI), first with a publicly available data set and then using it to validate the training process of a small dataset of acquired data. The first part of the article describes the methodology for acquiring EEG signals that imitated the information found in the publicly available dataset Physionet Motor/Imagery. The second part compares the training process for NNs. The first NN is a Bidirectional Long-Short Term Memory (BiLSTM) trained from scratch with the Physionet dataset, and the second NN is a CNN called SqueezeNet trained following the TL method with the small acquired dataset, reaching an accuracy of 91.25% in the BiLSTM with the scratch method and an accuracy of 92.33% with the transfer learning method for the EEG MI signal classification.

INDEX TERMS BCI, EEG signal processing, machine learning, transfer learning, motor/imagery.

I. INTRODUCTION

Electroencephalography (EEG) is the study of electrical brain activity. It is measured through electrodes placed on the scalp surface and then recorded [1]. These electrical signals are associated with communication between neurons and different body parts [2], depending on electrode position in the scalp associated with a brain region, a specific brain function, or neuron connectivity. These records also could reflect the activation of neurons during the execution of a function or the response to external stimuli.

Because of the many advantages of the EEG signals [3] like cost and their capacity to represent the neurocognitive process, this study is gaining space in developing the Brain-Computer Interfaces(BCIs). BCIs are systems that allow the connection between a person and the environment using the analysis of the electrophysiological response through the acquisition and encoding of physiological [4] signals like Electroocoulography (EOG), Electromyography (EMG), Electroencephalography (EEG), etc. The use of BCI technology is increasing with applications in a wide field of action like gaming, arts, and health applications [5]. In the BCIs for health applications, some fields that benefited from their use are the fields of diagnosis and rehabilitation. Diagnosis or Medical Diagnosis is a task related to the process of characterization of the signs and symptoms [6] and causes human diseases for their subsequent treatment. On the other hand, rehabilitation is a collection

The associate editor coordinating the review of this manuscript and approving it for publication was Nuno M. Garcia.

of clinical methods focused on offering solutions to several disability conditions and improving the body functions that could contribute to a person's social re-integration [7]. These two concepts are closely related because rehabilitation is in constant need of professionals for diagnosis and assistance to revise the medical procedures taken to restore the patient's functions and improve their quality of life quality, and the application of good rehabilitation measures allows the evolution of a medical diagnosis and the treatment of the health conditions. For example, these systems are used in upper limb rehabilitation [8], helping patients to visualize their progress through the recording and display of EEG signals in computerized systems, promoting the continuity of the treatment and avoiding abandoning therapy.

The analysis of EEG signals possibilities to find trigger signals for BCI technologies because EEG reflects the mental state and can capture the brain dynamics [9] like the difference between imaginary and real motor movement patterns. Motor Imagery (MI) are signals related to the cognitive process (imagination) of executing a movement (such as grasping or stomping) without the muscular activation [10].

Nowadays, the use of Machine Learning (ML) algorithms has made the signal analysis task easier. These algorithms apply statistical techniques using computational tools to predict values of a function [11]. This estimation is known as ''learning'' and can do diverse tasks, as mentioned before, like classification, clustering, ranking, etc., and is useful to cope with a large amount of data. According to the input information and their designated task (type of estimation or learning), these algorithms can be divided into supervised, unsupervised, and reinforcement learning [12]. Examples of these ML algorithms are Neural Networks(NN) [13], Linear Discriminant Analysis [14] Decision trees (DT) [5], Support Vector Machines (SVM) [15], Kean-Nearest Neighbor(KNN) [5].

Of all the previously mentioned algorithms, NNs have gained popularity in the rehabilitation field [16] due to their ability to generalize information (prediction) with large size of training data, their training time, and accuracy compared with other ML algorithms. The NNs are algorithms inspired by the structure and function of biological neurons, imitating the interconnection to process information. Each NN consists of nodes (computational units) arranged in layers (commonly the input, hidden state, and output layers) interconnected with each other. The connection is called weight, and the numerical value of the weight reflects the relation between the input and the output. The sum of the operation between the weight and the input is evaluated by a function(like the sigmoid function) that contains a threshold, depending on the value of the comparison between them, which will be the generated output.

The process when the networks are pondering the dependencies between the input, the threshold, and the output is known as ''training''. After this training process, the NNs can calculate the weights of the inputs that will trigger a response; this is the prediction phase. NN possesses different architectures with different characteristics, advantages, and disadvantages. For example, Convolutional Neural Networks (CNN) are NNs that possess a layer with a convolution operation. The convolution operation is the application of a filter matrix after the input layer that will be tuning in the training phase; this type of operation has had good results in image processing [17]. On the other hand, Recurrent Neural Networks (RNN) are networks that possess the capacity to give context to the output by creating a cycle between the input and the output and are widely used in machine translation (an automatic translation of a text) from their capacity to process information from a sequence (such as sentences) [18], making predictions more accurate. A sub-type of RNN architecture can also ''store'' information by having a memory block or memory cell; these are the Long-short-term memory (LSTM) NNs. This RNN architecture is distinguished from others by having a memory block that allows them to ''remember'' information, which made it useful for recent works in EEG classification [19].

One technique that helps to cope with problems of Machine Learning where the input data could be insufficient to feed an algorithm (because ML algorithms are greedy) is the Transfer Learning technique. Also called Inductive Transfer, Transfer Learning (TL) is defined as ''a situation where what has been learned in one setting is exploited to improve generalization in another setting'' [20]. Transfer Learning enables reusing existing knowledge of a previously trained algorithm for a new related task like classification or pattern recognition [21]. This technique has advantages like improving the baseline performance, reducing the overall learning time because the model does not learn from scratch but for a pre-trained model, and improving the final performance. For example, at [22], is described as a transfer learning technique used to train a deep learning model called EEGNet [23], which is described as a compact CNN focused on EEG-based BCIs. At this work, the CNN is re-training with two small datasets, the BCI Competition III IIIa and the BCI Competition IV. The first of the two datasets contains four classes of MI EEG signals of left and right hands, feet, and tongue movements for three subjects. The second dataset is a classical dataset involving 4 MI samples of nine subjects' left and right hands, feet, and tongue movements. For the obtaining of features, the Filter Bank Common Spatial Patterns(FBCSP) was used, and lasso regularization was used for regularization. The results for the classification task of this work were 88.89% average accuracy for the BCI Competition III IIIa and 81.34% average accuracy for the BCI Competition IV. Another work involving the transfer learning process is the one found at [24]; this work describes using the BCI Competition IV 2a dataset containing MI information for the classification task. To this set it was applied spatial filtering, frequency filtering, a data augmentation process, and a Continuous Wavelet Transform(CWT) were before fed the implementations of three hybrid models of NN's: the first is a CNN/LTSM with

a final accuracy of 86%, the second a Resnet-50-LSTM with an accuracy of 90% and finally an Inception v-3-LSTM with an accuracy of 92%.

The present work discusses the classification of MI EEG signals using two different NNs, a Bidirectional Long-Short Term Memory(BiLSTM) NN and a CNN architecture called SquezeeNet, with two training techniques: training an algorithm from scratch and using the transfer learning process. Also, the difference in their performance is discussed using the classification of a small dataset of MI signal information acquired in the present research as a validation method.

The paper is organized as follows: Section II provides information about the Materials and Methods used in the present. This section describes the publicly available dataset used for training, and the process followed to generate our dataset with a limited number of subjects. Also describes the methodology for pre-processing, feature extraction, feature selection, normalization, and the architectures and characteristics of the classification algorithms. Section III exposes the present results, remarking on the differences in various performance metrics of the different ML algorithms implemented used for the MI EEG classification and the differences in the training process (scratch and TL). Section IV discusses the impact of the present results according to the founding at state of the art and the recommendations for further studies analysis. Finally,V presents some conclusions derived from the present and outstanding points inferred from the performance metrics.

II. MATERIALS AND METHODS

Because the EEG signals are the reflection of the continuous and oscillating brain activity, the signals need to be processed and analyzed to capture this non-stationary information. The pre-processing allows the EEG signals to be easier to read and impact the performance of the algorithms [25]. Also, the methodology followed for the feature extraction is relevant to the class (class in ML is a label that is assigned to a data cluster that has similar characteristics)composition for the different ML algorithms and the classification process. Because the ML algorithms work through the generalization of new examples, the use of two different sources of data allows us to verify the learning process of the algorithms and also the methodology for the acquisition of signals. This section will describe the different followed processes (that can be seen in Figure 1) consisting of the description of the datasets, the pre-processing done at the signal, the features extracted, the selection of the information relevant to the system, and the classification achieved using different algorithms and techniques, to validate the learning process of ML algorithms, with the acquisition of new data that could be used as new information for the generalization process of the algorithms.

As materials, the present work used a Dell Inspiron 15 3000 with an AMD A9-9425 processor, two cores, a Radeon

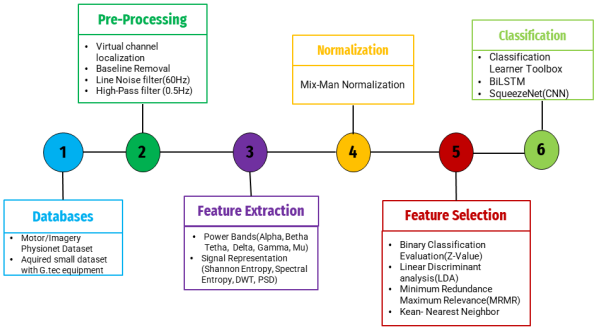


FIGURE 1. Flowchart of the methodology followed for EEG MI signal classification at the present.

R5 graphic card, a RAM size of 8GB and DDR4 SDRAM with a Windows 10 operative system, and an RB2020a version of the Matlab software. Also implemented some algorithms using the Python Anaconda distribution and the Spider IDE that has pre-loaded many scientific packages like Matplotlib, Numpy, etc.

A. DATASETS

This subsection will explain how are composed the datasets used in the present. Because EEG MI information is related to the brain regions associated with the movement, six EEG channels that could reflect this information were selected. Figure 2 highlights in the 10/20 system of electrode positioning the selected channels (Fc3, Fc4, Cz, Cpz, Fcz, and Fpz) according to their position in the sensorimotor area of the brain associated with the motor activities.

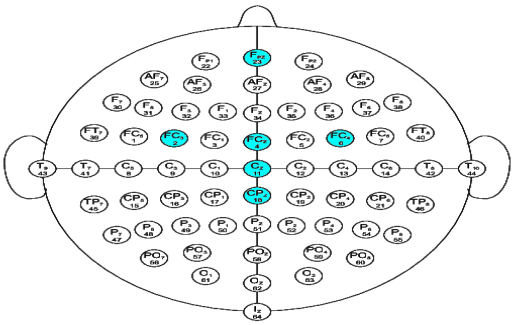


FIGURE 2. 10/20 Electrode positioning system with selected channels highlighted. These channels were choose according to the motor area associated with movement activities in the present.

1) PUBLIC DATASET

In the present is used, the Physionet Motor/Imagery Dataset [26]. This public dataset used in previous works like [19], [27], and [25] contains records of EEG signals. As previously mentioned, these records were taken using the BCI2000 software and 64 channels. These records consist of 1500 one and two-minute EEGs obtained from 109 subjects. Each subject performed 14 experimental runs (or trials): two one-minute baselines (first with open eyes, then with eyes

closed) and three two-minute length runs of each of the four tasks:

- Task 1 (open and close left or right fist).
- Task 2 (imagine opening and closing left or right fist).
- Task 3 (open and close both fists or both feet).
- Task 4 (imagine opening and closing both fists or both feet).

Figure 3 represents the final composition of the dataset, which includes 109 experimental subjects, 14 trials of the four tasks with three repetitions for movement, and imagination of the movement of two minutes of duration each.

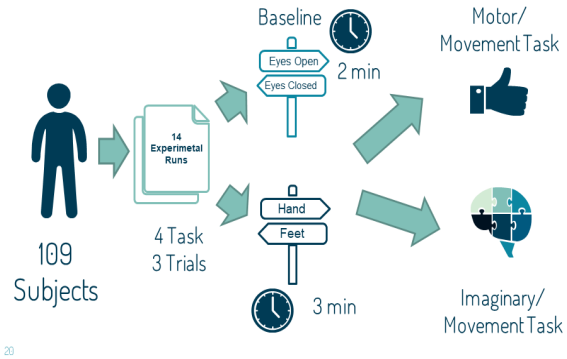


FIGURE 3. Public dataset (Physionet Motor/ Imagery) composition of the full dataset divided by task, trials, and subjects.

2) GENERATED DATASET

This data set was acquired by recording EEG signals using a system for the g.tec company called g.GAMMAsys. This system is designed to record EEG signals through a cap with active, non-invasive electrodes. This cap is connected to a g.GAMMAbox, which is the link between the electrodes and the biosignal amplifier (g.USBamp). Some of their technical specifications are that the equipment possesses a battery supply of 9V (from a 110 V of AC and a 60 Hz power line), a DC Filter of 10 kHz, etc. The g.GAMMAbox used in the present allows for connecting up to 16 different electrodes, but only six were used with a reference channel. These six channels were previously mentioned and found in Figure 2. This system was selected for the medical grade records and its ability to reduce the artifacts in the records, the fast electrode montage, and the multi-channel recordings.

The present dataset was obtained under the Helsinki Declaration [28] and with the signed consent of the participants in the study and during two days of trials. First, the cap was allocated with the electrodes in the volunteers' scalp. This cap is connected to the previously mentioned g.GAMMAsys (g.GAMMAbox and the amplifier) and the system are connected to a computer to save the records using the g.tec software. Once the equipment was placed, the trials were realized. Four subjects performed eight experimental trials of one-minute recording with open eyes and then one minute with eyes closed and three trials of two minutes length of each of the two tasks:

- Task 1. An image appears on the screen and indicates the movement of the right, left fist, or both fists with a color and a word. The subject opens and closes the corresponding fist or fists until the target disappears. Then, the subject relaxes. (Open and close left or right fist.)
- Task 2. An image appears on the screen and indicates with a color and a word the imagination of the movement of the right, left fist, or both fists. The subject imagines opening and closing the corresponding fist or fist until the target disappears. Then, the subject relaxes. (Imagine opening and closing left or right fist.)

At [26], it was described that the stimulus to perform the MI was a visual stimulus, so the subjects were conditioned to perform or imagine the movement under visual stimuli. The obtained EEG records have a sample rate of 256 Hz.

Because the EEG records obtained were few and to secure the performance of the ML algorithms, it was decided to record three minutes of the movement task and then do a data augmentation process, where the signal was segmented according to the following:

- Fragments of 45 seconds of the signals of movement and imagination of the movement
- Fragments of 5 seconds of a duration of the signals that correspond to the Baseline

With the data augmentation process, the available information for the ML feed is comparable to taking a sample of 10 subjects from the Physionet dataset. Figure 4 represents the composition of the data acquired with a total of 8 trials (two for Baseline, three of three minutes of real movement, and three for imagination of movement) per subject.

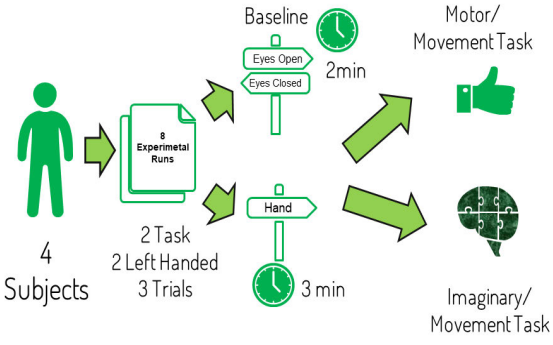


FIGURE 4. Acquired dataset composition of the dataset divided by task, trials and subjects.

B. PRE-PROCESSING

The pre-processing stage was carried out using the same steps for both datasets, using the EEGLab plug-in for Matlab. The processes applied to the signals were the following:

The processes applied to the signals were the following:

- Virtual location of the corresponding channels under the 10/20 system.
- A baseline removal for reference purposes was used.

- A noise filter was applied to deal with the noise that could be found in the original signal. The Noise filter is a digital bandpass filter that has a frequency related to the frequency of the electrical supply; in the North American area where the signals have been acquired(both the Physionet Dataset and the Generated Dataset), this frequency corresponds to approximately 60Hz, in another world region, could be for example 50Hz like in Europe.
- A High-Pass filter, with a value of 0.5Hz range of cutoff, was used to cope with frequencies that could interfere with the baseline.

C. FEATURE EXTRACTION

For the feature extraction and feature selection tasks, previous works explore different approaches, like the use of the pre-processed signal with statistical algorithms for feature extraction [29], the premeditate absence of a feature selection task [30], and the use of different time-domain characteristics in combination with feature selection algorithms like [31]. The present approach comprises the following processes: the pre-processing of the previous data acquired (Physionet Dataset and Acquired dataset), feature extraction (with a time-frequency and entropy approach), and feature selection (with filter approach) before a classification process with the NNs. These NNs feed with numerical values like [32] and graphical information like [24].

1) POWER BANDS

The EEG records have an amplitude of approximately 100 μV and a 50-100 Hz range. This second value reflects the information at the frequency domain, and by examining this domain, it could be assessed the state of brain activity [4]. Frequency bands are the interval of energy at that domain. These intervals have an upper limit and a low limit, and the information that could be reflected depends on these limits. These bands are Delta (work below 4 Hz), Theta (4-7Hz), Alpha (8-13 Hz), Beta (14-50 Hz), Gamma (above 30Hz), and work below from 4Hz to above 30Hz [9].

At [33], it is indicated that the bands that are involved in the MI are the alpha, beta, and gamma bands. The present work also extracted information on an alphoid band (in the range of the alpha band) called Mu because this band is related to motor activity and the imagination, realization, and observation of both motor activities and motor intentions.

At present, the Average Band Power is calculated as one single number that summarizes the given frequency band's contribution to the signal's overall power.

2) SIGNAL REPRESENTATION FEATURES

One of the most important processes in the EEG signal analysis discipline is the one related to the features that could be obtained to reflect the information of a signal. Previous studies describe the interaction between brain regions [34], others on the interaction of brain and physical stimuli [35], etc. The feature extraction process allows converting a raw signal into numerical features that make it easier to process the great amount of data in the datasets and improve the performance of ML algorithms, which helps reduce the dataset without losing information. There are many techniques and approaches for feature extraction according to the domain of the information that will be discussed. The common domains found in the literature are the time domain, the frequency domain, and the combination of the two called time-domain techniques [36].

The present study extracted five features under the domain criteria: Power Spectral Density (PSD) is under the spectral information as the Spectral Entropy and the Shannon Entropy, and two Time-Frequency transformations using a Discrete Wavelet Transform (DWT) and Spectrograms.

Firstly, PSD is defined as the rate at which motor units are fired. Many other features can obtained by applying a mathematical analysis of PSD [36]. The mathematical expression of PSD is given for Equation 1 that is a representation of the Discrete-time Fourier Transform (DTFT) of the auto-correlation sequence of the signal, that is one of the definitions of the PSD:

$$\phi(\omega) = \sum_{k=-\infty}^{\infty} r(k)\, e^{-i\omega k} \tag{1}$$

where $\phi(\omega)$ is the average signal power over the frequencies being ω a real number that represents the frequency of the signal, $\sum_{k=-\infty}^{\infty}$ is the condition that needs to be fulfilled to be a discrete-time sequence that establishes that the sequence is a discrete-time sequence only if is summable, $r(k)$ is the autovariance function or the function that allows to calculate the co-variance in different periods where k is an integer that has units of cycles/samples and $e^{-i\omega k}$ is the approximation of the Discrete-Time Fourier Transform of the signal.

Entropy is a measure of uncertainty [37]; this uncertainty is translated as the number necessary to represent information, in the present case, the number of bits necessary to represent the signal. This measure possesses many approaches for its calculation. In the case of Shannon Entropy, this calculation combines the wavelet decomposition with a measure of the coefficients by a determined scale and is widely used for non-stationary signal analysis like EEG. The calculation is determined by a mass function observed at Equation 2.

$$x_i = \frac{x_i}{\sum_{i=1}^{N} \cdot x_i} \tag{2}$$

where x represents a discrete random variable, i the possible outcomes in the probabilistic space, and $\sum N_{i=1} \cdot x_i$ represents the average amount of information that could be represented in the same considering all the expected outcomes.

Another calculation of the entropy is called Spectral Entropy [37], which is a measure of the signal's spectral power distribution based on the Shannon entropy calculation, where the normalized power distribution in the frequency domain is the probability distribution. This is calculated by Equation 3, and the obtained value can be used as an estimate for voicing/unvoicing decision; it is expected that sub-bands

that are flatter will have higher entropy and the sub-bands where a formant will have low entropy.

$$H = -\sum_{i=1}^{N} x_i \log 2 \cdot x_i \tag{3}$$

where the sum is taken over all values that the random variable can take.

A DWT is the decomposition of the signal into a set of basis functions based consisting of contractions, expansions, and translations of a mother function Φ(t), called the wavelet [38]. This type of transform [39] aims to represent a discrete-time series as a set of coefficients. This transformation can be represented as a filter back of one high pass and a low pass component. The equation that relates to this is:

$$d_k = (-1)^k C_{L-k} \tag{4}$$

where *dk* represents the row vector of signal information *C* is the applied filter vector, and the operation expresses the multiplication necessary for a vector transformation.

The spectrograms representing information at three axes [40] in an image were taken to feed the TL algorithm, which, as previous works like [24], is a CNN.

The features previously mentioned were extracted for four tasks at present: two tasks related to the real movement of the upper limbs (hands) and two for the imagery movement. Each of these tasks was repeated for three trials of each dataset's subjects, the public one and the generated one. Figure 5 represents the extracted features for the Physionet dataset, with an approximately final number of over 72,000 features to classify with 36,000 points for each of the proposed classes (movement and inactivity).

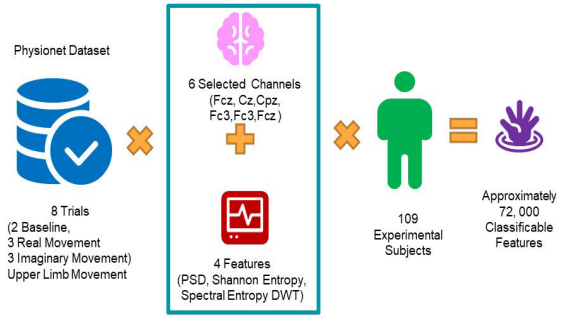


FIGURE 5. Composition of the physionet dataset in terms of trials and extracted features.

D. NORMALIZATION

Normalization is used in previous works like [41] to improve dataset management by identifying relationships between variables within the same dataset. Different types of normalization exist; at present, the most common one is max-min normalization. In this normalization, the greatest value in a group is transformed into a one, and the smaller value into a 0. This is a re-scaling and changes the distance between the min and max values without losing the mathematical distribution and relationship between data. The Equation5 represents the re-scaling process for data to any interval (in this case [0, 1]), where [a b] is the interval to re-scale, and this allows to preserve the Z-scores of the data.

$$X_{Rescaled} = a + [\frac{X - \min_X}{\max_X - \min_X}](b - a) \tag{5}$$

where X is the value to normalize, min_X is the smaller value of the cluster, and max_X is the maximum value of the cluster found in the database.

E. FEATURE SELECTION

The Feature Selection process, also called "dimension reduction" [36] allows to establish the relationship between features (independence or co-dependency) and outstand the ones that could apport more information to a class or are more probably- like that improve the models. It has been seen in numerous previous studies (for example [42], and [33]) that using system-important features improves the performance of various ML algorithms in the training stage and reduces computational time. Because the Physionet is a high-dimensional dataset that will be compared with the information of a small dataset, it is necessary to search for that information to improve the performance of the ML algorithms and reflect the relevant information for the classification process. For dimensionality reduction, different approaches exist (like genetic algorithms, subset selection, the manual filter, the correlation through indices, etc.); one of the most used approaches is metaheuristics (an optimization procedure that uses heuristics as a search basis). This approach contains methods that are focused on reducing the irrelevant information to classify (the information that is not related to the goal of the classification)and outstand the information that could perform better, using algorithms that can compute the relation between the features in the high-dimensionality sets. We currently select the filter and metaheuristic approach to deal with the Physionet dataset and its high dimensionality.

Before starting with the ML algorithms for feature selection, an independent evaluation criterion for binary classification was implemented to select a pair of features whose evaluation performs better in binary classification. This evaluation was done through the calculation of the correlation information of the Z-value, which could be found in Equation 6 this value is in a range of 0 to 1 when a higher value expresses statistic significance, where Z corresponds to the Z-value or Z-score, ρ is the average value of the cross-correlation coefficient between the candidate feature and all previously selected features, α is the weight value that sets the weighting factor. If the value of ρ is close to 1, that means that the selected values are correlated between them.

$$\text{Feature Weight} = Z \times (1 - \alpha \times \rho) \tag{6}$$

The algorithms selected for this task were Minimum Redundancy Maximum Relevance (MRMR) [43], Linear Discriminant Analysis [14], and Decision trees [5]. MRMR

is an algorithm that reduces redundant data within the dataset and maximizes the relevant information to classify using the following mathematical formulation that could be seen in Equation 7 where I represents the mutual information, S is the set of features, V_s is the relevance of the set and W_s is the redundancy of the mathematical probed features.

$$V_s = \frac{1}{|S|}\sum_{x\epsilon S} I(x,y), \quad W_s = \frac{1}{|S|^2}\sum_{x,z\epsilon S} I(x,z) \tag{7}$$

The Matlab implementation of this algorithm ranks all features and returns the indices of features ordered by feature importance.

The Linear Discriminant Analysis, on the other hand, is the evolution of a statistical method designed to distinguish between classes of plant [14] and is defined by the probability of the joint discrete/categorical variables on their distribution. The Equation 8 where $\widehat{y}$ is the predicted class, K is the number of classes, $P(k|x)$ is the posterior probability, and $C(y|k)$ is the cost of classification of each observation, represents the way the algorithm is applied to create a hyperplane that could separate two categories of features, remarking the features with better performance is the classification task.

$$\widehat{y} = \arg\min \sum_{k=1}^{k} \widehat{P}(k|x)C(y|k) \tag{8}$$

And the KNN algorithm classifies new values based on similarity [5]. This independence is calculated through distance metrics, like Euclidean distance or cosine, etc., to find the closest feature. This similarity is achieved using two centroids, and the distance is measured for all the observations. This is how the cluster is made; this centroid moves until the process ends.

F. CLASSIFICATION ALGORITHMS

1) LONG-SHORT-TERM MEMORY

A Long-Short Term Memory or LSTM is a type of Recurrent Neural Network (RNN) (architecture at Figure 6). The RNNs are NNs that have a circulation behavior through calculations (recurrence), meaning that their output depends on the current input and the previous output. This characteristic brings context to a classification task [11].

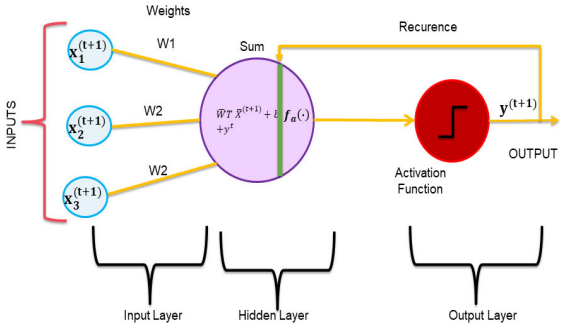


FIGURE 6. Recurrent neural network architecture.

The LSTM [44], comparable to the RNNs, has a recurrence that allows processing information like the EEG sequences. This recurrence operation is done by memory cells called memory blocks(representation at Figure 7). These Memory cells or blocks are composed of an input gate, forget gate, and output gate, which are in charge of preserving previous output states using their feedback connections to store representations of recent input events and using a threshold to update information controlling the flow of information inside the cell block.

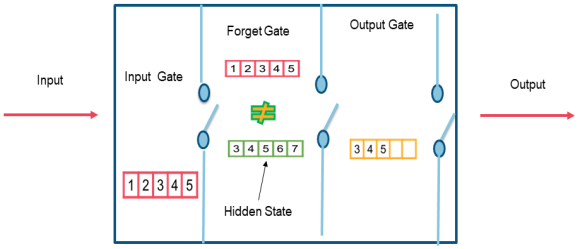


FIGURE 7. Memory block of a LSTM.

The input gate decides the information to store in the block (Equation 9), the forget gate (Equation 10) using the threshold is in charge of the decision of preservation or not of the information, passing through the previous hidden state and the activation function and the output gate (Equation 10) controls the pieces of information to conserve using the previous information. At the Equations 9,10,11, the symbols represent: σ, W_{xi} represents the weight value in the input, output or forget gate is the letter in the subscript ("i," "f" or "o") the distinction, X_t is the vector that alludes to the skip in the time, h_t is the hidden state vector, c_t is mentioned as the cell state vector and b_i the bias that changes through the learning process.

$$i = \sigma(W_{xi}X_t + W_{hi}h_{t-1} + W_{ci}C_{t-1} + b_i) \tag{9}$$
$$f = \sigma(W_{xf}X_t + W_{hf}h_{t-1} + W_{cf}C_{t-1} + b_f) \tag{10}$$
$$o = \sigma(W_{xo}X_t + W_{ho}h_{t-1} + W_{co}C_{t-1} + b_o) \tag{11}$$

The LSTM architecture has a sub-type called BiLSTM, whose most important change is the ability to learn the information not only in a forward direction but also in a backward direction, which is useful for learning complete time series at each time step. The layers' characteristics can be seen in Table 1. Besides the Bidirectional LSTM layer, the architecture used contains a fully connected layer and a softmax layer, the last one to deal with the gradient descent problem.

TABLE 1. Characteristics of the Layers at the implemented BiLSTM.

Layer	*Value*
Input Layer	Number of Features (2 Sequences of 1200 values)
BiLSTM unit	BiLSTM with 100 hidden units
Fully Connected	2 (Corresponding 2 classes)
Softmax	Softmax operation
Classification Output	2400 Sequential Data

2) CONVOLUTIONAL NEURAL NETWORK WITH TRANSFER LEARNING APPROACH

At Transfer Learning, the goal is that the learning acquired in an algorithm is intended to be used in a new training process or for a new task. For this technique, there are different methods; for example, in inductive transfer learning, the source and target domain are the same but have different tasks. In Unsupervised Transfer Learning, the source and target domain data are not labeled, and finally, in Transductive Transfer Learning, the source and target tasks are similar, but their domains are different with a source domain that possesses a lot of labeled data, but the target domain possesses no one.

This work uses a pre-trained CNN model available at the Matlab Library. A CNN is an NN architecture that relies as its name indicates on the convolution operation.

The used CCN model is called Squeeze Net [45]. The SqueezeNet is comprised of 18 layers and is trained with a million images from the Image Net database. This is a CNN intended to preserve the accuracy of another model called AlexNet [46] with fewer parameters replacing 3×3 filters of the AlexNet model with smaller 1×1 filters, decreasing the number of input channels and downsampling late to the convolution layers having a large activation map.

The last layers must be modified before re-training the net with the new information or data set. The layers of interest are the last convolutional layer (that in Matlab is called conv10), which in this net model replaces the Fully connected layer and the classification layer (Classification Layer_predictions at the present). These layers need to be replaced by new layers that contain parameters that follow the training options for the new data and the learning process faster than in the transferred layers. At the convolutional layer, the parameters that are suggested to modify are the filter size to supervise that has a size that will not delay the training phase, the number of filters that need to be set as the number of the classes to train in the present this value needs will be set to three, that corresponds to the three available classes that in this case is the identification of the baseline, the real movement or imaginary movement, the Weight Learn Rate Factor that changes according to the classes learned, to a major number of classes this number needs to be bigger instead in this occasion this number is set to ten as same as the Bias Learn Factor.

III. RESULTS

This section will expose the different results in feature extraction, feature selection, and classification with the different algorithms chosen to classify the MI information.

A. FEATURE EXTRACTION

As mentioned, ten features were extracted: 6 power bands and four features with Spectrum and time-frequency domain information. These features will be the input of the LSTM algorithm.

1) CHANNELS AND POWER BANDS

Each Physionet dataset trial and the signal recording using g.tec extracted 6 Power Bands with the Average Band Power calculation: Alpha, Beta, Theta, Gamma, Mu, and Delta.

Figure 8 compares the Alpha Power band with the non-filtered signal of one trial of the real movement task.

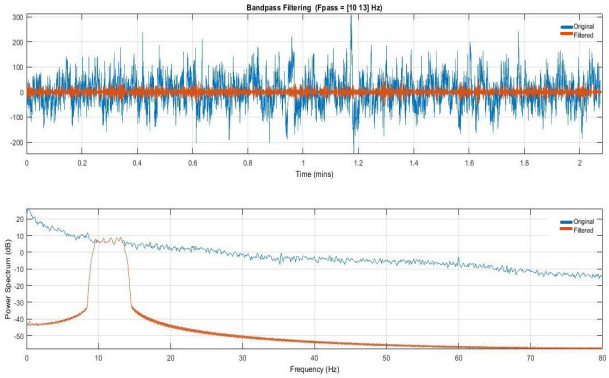


FIGURE 8. Alpha power band of real movement trial of subject 1.

2) SIGNAL REPRESENTATION FEATURES

- PSD. Figure 9 shows the PSD calculation for a trial of one of the three tasks: Movement of the upper limbs, Imagery of the movement, and the Baseline. This calculation is the normalized version, which implies that the frequency was changed in their scale, dividing their original frequency of 256Hz by itself to have a scale of cycle/sample (in the present, a radian/sample scale) and to express the power (at db) of the signal. The image compares the power distribution at each signal's frequency domain, pointing out that the Movement and Imagery PSD have similar frequency components. Previous works of EEG classification as [33] mentioned a correlation between imagery and motor signals and their contribution to the performance of a classification task [47]. The relationship between Movement and Imagery is close to compound the Movement class.
- DWT. The Daubechies family was chosen for its calculation of this feature. This family was chosen because it is a family of orthogonal wavelets of compact support that facilitated the analysis of discrete wavelets, particularly for the detection of features (like in the present) and for the elimination of noise. Another reason is that the one that presents the highest number of vanishing moments, at present, the db5 parameters reflect the number of vanishing moments(five) because it is useful for time and frequency localization. In Figure 10, it can be appreciated that the original signal and the signal with the DWT. The reconstruction with DWT is shorter in the time domain but has the same information in the frequency domain.

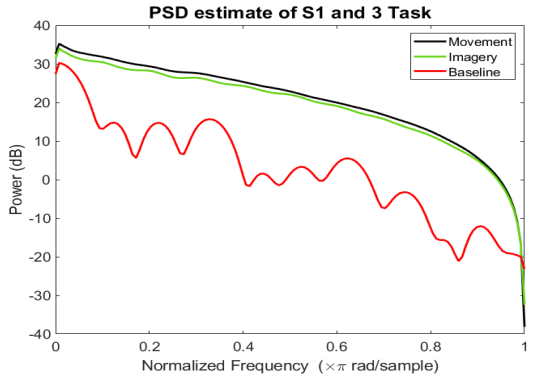


FIGURE 9. Comparison of the power spectral density for one subject and baseline, real movement and imagery movement.

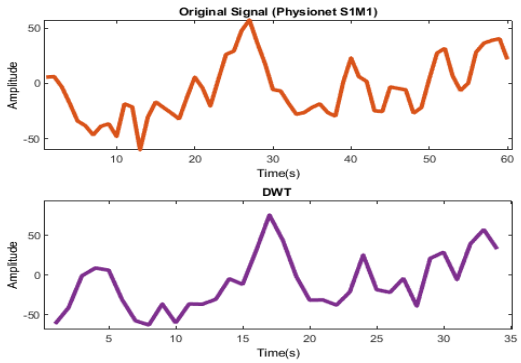


FIGURE 10. DWT comparison for one trial of movement task a) original signal for one subject of movement task b) DWT for one subject and movement task.

- Spectrogram. The Spectrogram was calculated because it reflects the frequency spectrum information versus the time in an image [48]. Figure 11 represents a comparative of the Spectrograms of 1 subject for Real movement of the two available data sets expressed in samples vs. frequency.

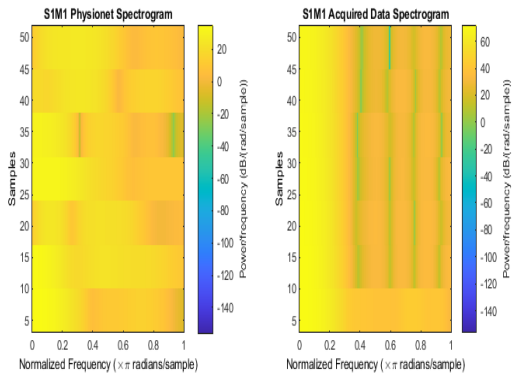


FIGURE 11. Spectrograms of real movement task for the first subject a)spectrogram of the physionet data set b)spectrogram of the acquired dataset.

- Shannon Entropy. Applying the Equation 12 where $H(x)$ is the entropy of a discrete variable, P is the probabilities i is the scale, and the sum is the expression of all the values that the random variable can take, is calculated one single value that expresses the entropy measure in the signal.

$$H(x) = -\sum_{i} P(X = x_i) \ln\left(P\left(X = x_i\right)\right) \tag{12}$$

At Figure 12 is represented the obtained values of the Shannon Entropy Features for the Fc3 and Fc4 channels of the all available data.

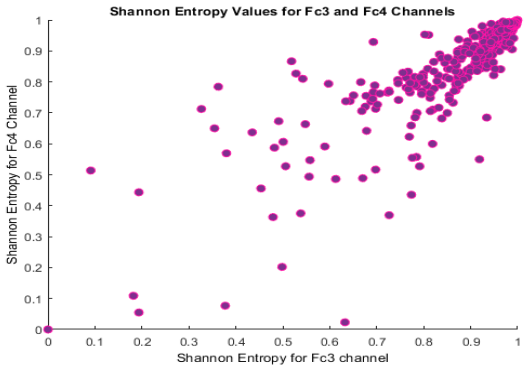


FIGURE 12. Shannon entropy Fc3 and Fc4 channels.

- Spectral Entropy. In the case of Spectral Entropy, it was calculated for each trial. The algorithm was fed the mean of each signal spectra. Figure 13, as an example, compares the original signal for a Baseline trial and their Spectral Entropy calculation.

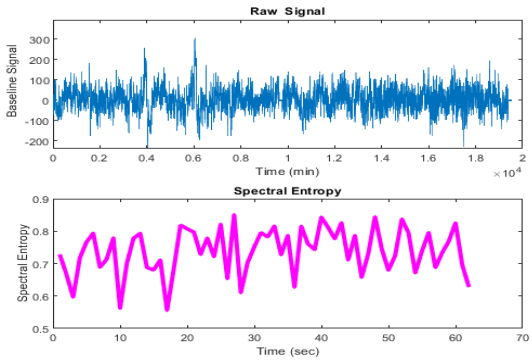


FIGURE 13. Spectral entropy comparison of one trial of baseline (eyes open eyes closed) a)Original signal of a trial of baseline b)Spectral entropy of the baseline task.

B. FEATURE SELECTION

For the feature selection process, the first implementation was the raking of the independent evaluation criterion for binary classification, then was implemented ML algorithms (MRMR, LDA, Decision Trees, and KNN)in Matlab and Python environments to revisit whether the change of the implementation platform will change the outstanding features for the system or give slight variations in the

data. These algorithms were fed with the total amount of obtained characteristics (approximately 72,000) of the Physionet Dataset; this task ranked 60 features, ten features for one of the six selected channels into two target classes, movement and inactivity. The results of each and their platform or platforms of implementation are described as follows:

- Ranking evaluation for binary classification. As mentioned before, this implementation allows the rank of the correlation in the features, the independent features being the ones with a low score, the statistical independence features, and the ones that represent the system and have the best performance in binary classification. Figure 14 represents the first five ranked features. All the features were ranked, but passing the first three positions, the classification score dropped to a half, and passing the tenth position, the score dropped to values under 10. The first two positions correspond to the DWT for the Fc3 and the DWT for the Fc4 channels.

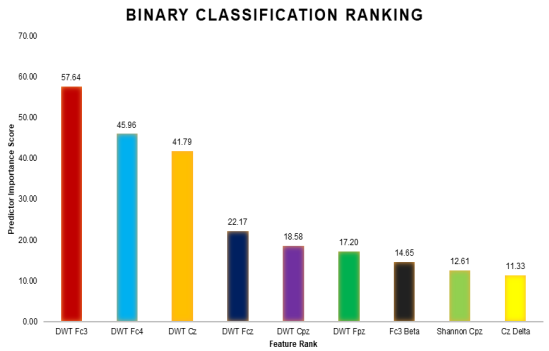


FIGURE 14. Binary classification ranking of the top 10 features.

- MRMR. The first ML algorithm that was used in the feature selection was the MRMR. Previous works like [43] showed that this algorithm is useful to rank the best feature in a system, which remarks the co-relation between different features, and this co-relation gives a score to the different tested features in a system. Figure 15 is the rank of the two best-scored features for the system, which remark the DWT for the Fc3 channel and the DWT for the Fc4 channel.

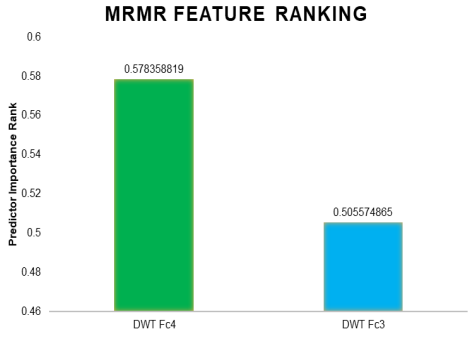


FIGURE 15. MRMR ranking (5 features) for the system.

- LDA. For the LDA implementation for feature selection, the features highlighted previously by the MRMR and Binary Classification criteria algorithms were probed with different combinations of the ten first algorithms. In this process, the two top features in the MRMR ranking and the binary classification criteria were the pair that worked efficiently with this algorithm. Figure 16 presents the classification performance with LDA for this pair of features corresponding to the DWT for the Fc4 channel and the DWT for the Fc3 channel, holding a 95.3% of classification accuracy.

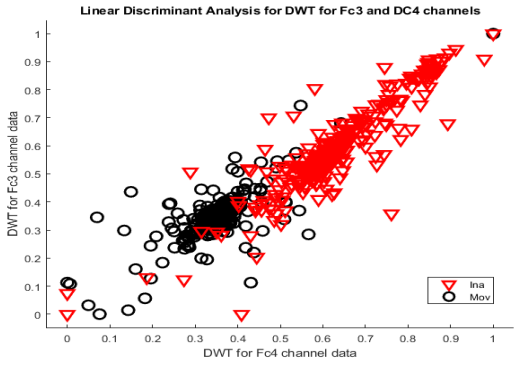


FIGURE 16. LDA Performing for Fc3 and Fc4 channels.

- Decision Trees and Random Forrest. In the case of Decision Trees and Random Forrest, both were implemented in two environments: Python and Matlab. It was decided to use the DWT of Fc3 and DWT of Fc4 channels pair of features according to the performance seen at the other implementation.
Figure 17 represents the percentage of Loss of the Random Forrest model implementation in Matlab according to the number of learning cycles used.

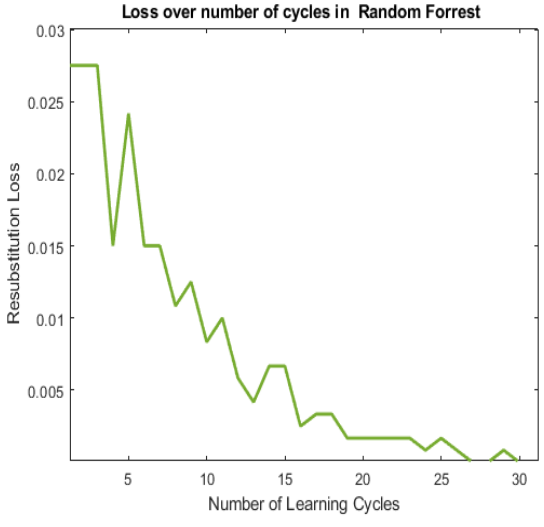


FIGURE 17. Random forrest loss according to the number of cycles for feature selection at matlab implementation.

The implementations were made in two different environments to verify that the variation of information y/o platform for implementation doesn't change the performance of the selected features or the performance of an algorithm. Figure 18 represents a Decision Tree classification for the two proposed classes with these features for Matlab.

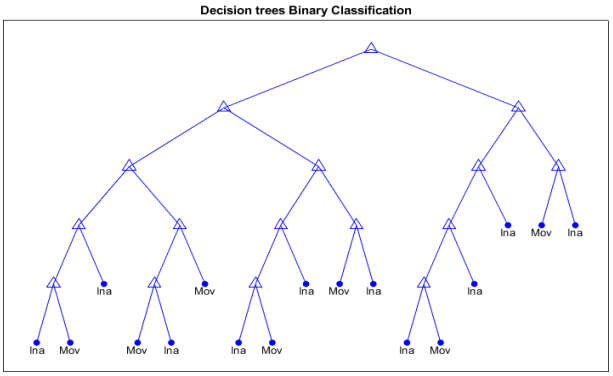


FIGURE 18. Decision trees classification in matlab implementation.

- KNN. This algorithm was also implemented in Python and Matlab to verify the classification process in the two platforms. The clustering of the features to the selected best pair performs similarly on both platforms. Figure 19 represents the decision boundary and the classification points for the KNN algorithm with the DWT for the Fc3 channel and DWT for the Fc4 channel features.

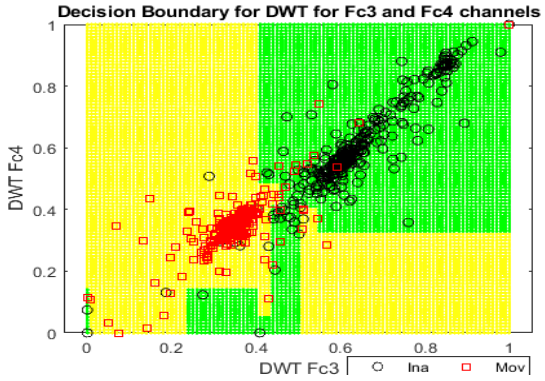


FIGURE 19. KNN decision boundary and classification points for DWT for Fc3 and DWT for Fc4.

At the four algorithms, the features that perform a high relevance score to the system were the DWT for the Fc3 channel and the DWT for the Fc4 channel; as other works remark [49], the DWT feature is useful for the classification task. Figure 20 compares the decision boundaries for the LDA, Decision Trees, Random Forrest, and KNN algorithms.

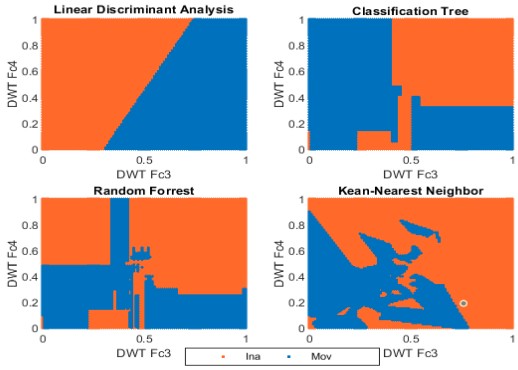


FIGURE 20. Decision boundarries for DWT for Fc3 and DWT for Fc4 features. a)LDA b)Classification trees c)Random forrest d)KNN.

C. CLASSIFICATION

1) LONG-SHORT TERM MEMORY

To compare the impact of the normalization process on the performance of the BiLSTM algorithm, it was implemented two kinds of BiLSTM, one with the remarked pair of features by the feature selection process (DWT for Fc3 and DWT for Fc4 channels) with normalization and the same features without the normalization.

Each of the different implementations has the parameters seen in Table 2.

TABLE 2. Parameters of BiLSTM network.

Parameter	*Value*
Gate Activation Function	Sigmoid
State Activation Function	Tanh
Optimizer	Adam
Learning Rate	0.01

In the training phase for the Classification Algorithms where used the cross-validation process. The implementation of the cross-validation process is to avoid the over-fitting of the algorithms. In the case of the Matlab environment, it uses the k-fold technique, at the present work, the value of K is five, selected by the size of the data set.

For the training phase of the BiLSTM NN, the data corresponding to the remarked information by the diverse algorithms in the dimension reduction or feature selection process, which is the DWT for channels Fc3 and the DWT for Fc4 channel, was used. This data corresponds to the mean value for each trial of the information of 100 subjects. For each subject, are available 12 trials, corresponding to 6 trials of Baseline, which is a segmentation of the complete Baseline to balance the information in the proposed classes (Inactivity and Movement), three trials for the Real Movement, and three trials for the Imagery Movement, that made a total of 12 trials per subject. For the testing set, the information corresponds to the same 12 trials for the nine subjects. The same algorithm was implemented twice, being the difference between the first and second implementations of the Normalization process; one is carried out with the extracted features without

the normalization process, and the other with normalized data.

The performance of the mentioned pair of implementations can be seen in Table 3. This performance is measured through the test set, is different from the performance in the training set, and was used as an indicator of overfitting.

TABLE 3. Performance of LSTM implementations.

Features	*Accuracy %*
DWT Fc3/DWT Fc4 With Normalization	91.25
DWT Fc3/DWT Fc4 Without Normalization	78.5

The best BiLSTM implementation (the one that carried out an accuracy of 91.25%) has a reported sensitivity of 83.33% with the classification information for the testing set. Figure 21 represents the Confusion Matrix for the best implementation of BiLSTM.

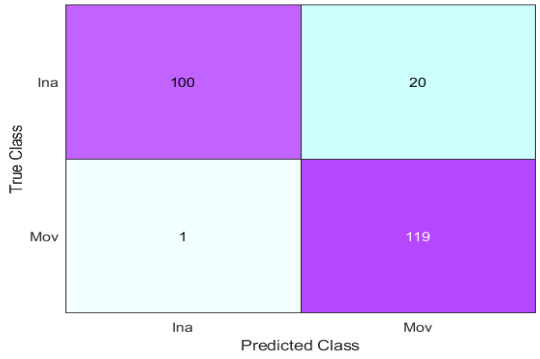


FIGURE 21. Confusion matrix for Fc4/Fc3 pair implementation of BiLSTM.

To verify if the information collected is also useful for doing a binary classification, used four subjects per 12 trials of the DWT of Fc3 and Fc4 of the acquired data set as a testing set of the BiLSTM trained with the 12 trials per 100 subjects with the Physionet dataset. Remarking that this NN was training with the information corresponding to the DWT for the Fc3 channel, DWT, and Fc4 channel. The results can be seen in Figure 22 with a reported accuracy of 53.98% and a sensitivity of 16.67%. This testing set is smaller than the testing set used in Figure 21 and with the reported accuracy below the median of the cases available, being a problem of generalization for the new cases.

2) TRANSFER LEARNING WITH SQUEEZE NET

The transfer learning classification process used a Convolutional Neural Network called Squeeze Net [45]. This CNN was fed with information on the newly acquired data to compare the performance of the new data with the Physionet and the ability to generalize the new SqueezeNet. This information comprises images corresponding to Spectrograms of segments of the original pre-processed signal.

The segments have a duration of 45 seconds for Movement and Imaginary conditions (the Movement class) with a total of 90 images. The Baseline (Inactivity class) is composed

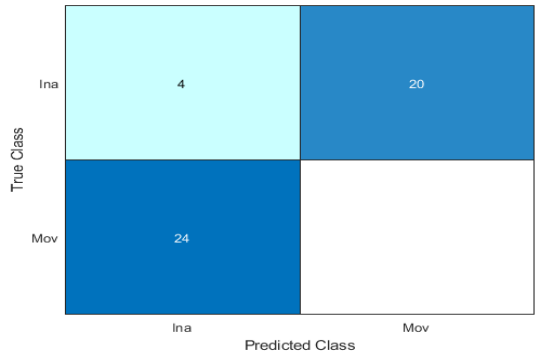


FIGURE 22. Confusion matrix for Fc4/Fc3 DWT of g.tec dataset acquisition.

of fragments of a duration of 5 seconds, with a total of 90 images. The testing set is composed of 10 images of both classes. To apply transfer learning, we re-train the NN with the new information. To achieve this, the last layers correspond to the last convolution and classification layers. Both parameters were changed to correspond to the binary classification (Movement and Inactivity classes). Once the change was made, the training data was divided into 70% training and 30% validation data.

Then the testing set was probed, with the results seen in Figure 23 for the acquired dataset, with a mean accuracy between predictions of 92.23 % and a sensibility of 83.33%. In the case of the SquezeeNet implementation with the Physionet dataset, it was found an accuracy of 60.6% and a sensibility of 83.33% in a test set of the same size as the acquired dataset implementation, having the Squeezenet implementation with the acquired dataset best performance in this case.

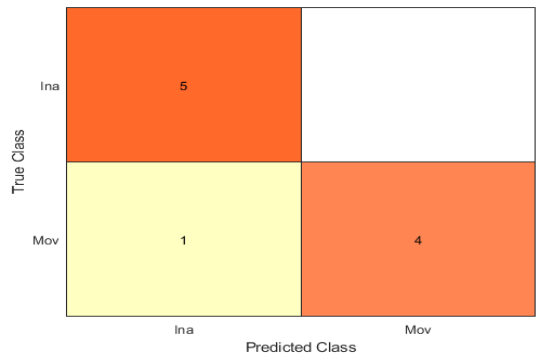


FIGURE 23. Confusion matrix for transfer learning testing set for acquired data.

In the case of Transfer Learning, previous works like [50] mentioned the difficulty of comparing the implementation of these algorithms because the input and the output domains are different. However, the two implementations currently have the same output domain and can compare the Neural Networks' performance, computational time, and available information. Table 4 referred to the differences between the

implementation, remembering the classes of the output are the same.

TABLE 4. Resume of classification algorithms details.

	Physionet	*G.tec*
Sample Size	100 subjects	4 Subjects
LSTM implementation features	DWT FC3/FC4 (Numerical)	
Squezeenet implementation features	Spectrogram of Pre-Processed Signal (Graphical)	
LSTM Accuracy	91.25%	53.98%
LSTM Sensitivity	83.33%	16.67%
Squezeenet Accuracy	60.6%	92.23%
Squezeenet Sensitivity	83.33%	83.33%

IV. DISCUSSION

A. TRANSFER LEARNING COMPARATIVE

Previous studies like the ones mentioned in [50] remark on the difficulty of comparing the performance of Transfer Learning methods because, in some implementations, the information that changes between the source information and the target information is the domain, so the original training information and the output information doesn't have enough similarities to be compared. This comparison is possible because both algorithms have the same output goal, and the training set is similar in their task and compoundings. As a result of the performance metrics, the present portrays a 92.23% accuracy for the binary classification at the transfer learning approach against the 91.25% accuracy in the BiLATM training, so this proves to be a good performance in comparison with studies like [51], and against similar classification task for the SquezzeNet and the BiLSTM.

B. SAMPLE SIZE

The size of the EEG dataset for the training of ML algorithms is always a limitation because ML algorithms are greedier, and the data reported for their training (quantity and quality) impact their performance, as seen at [31]. The size of the newly acquired dataset is impacted because the number of participant volunteers for the present work is small, so the direct training with the BiLSTM had low performance. Another inconvenience with the dataset was the restrictions of the publicly available dataset Physionet because some of their information is unavailable in their records, like the specific visual stimuli used to trigger the motor and imaginary task, so the Motor/Imagery tasks were adapted from the information portrayed by their protocol.

One of the future directions of the present study is to take more EEG records to have enough information to compare a dataset that could be compared with a sample of the Physionet dataset and improve the performance for LSTM (in the present new dataset performance for LSTM was 53.33%) because that reflects performance indicates the lack of information to train this type of algorithm. Also, implement other diverse TL algorithms for Physionet and the newly acquired dataset.

V. CONCLUSION

The contributions of the present works are the comparative performance of two approaches of ML algorithms for EEG MI classification and the acquisition of new EEG records to be compared and classified in them. Because one of the major requirements for the analysis of the EEG signals is their quality, another contribution of the present is the recording of the new data with medical-grade equipment that imitates the experimental protocol and the reproducibility of a publicly available EEG dataset. According to [52], the EEG acquisition systems with the use of gel in the electrodes (wet electrodes) placed on the scalp, like the one used in the present, possess a minor quantity of artifacts and better signal-to-noise radios and have good quality to be analyzed. Another advantage of using this equipment, according to [53], is that the user felt less discomfort and was more motivated to realize the different tasks.

Table 3 is portrayed a comparison of the performance of the different implementations; this includes the composition of each algorithm and the performance metrics as accuracy and sensibility. It is observable that the Transfer Learning approach has the highest accuracy with 92.23% compared with the performance of the BiLSTM with the same dataset as testing ser that only achieved 53.98% because, as mentioned in previous the literature, CNN algorithms have a better performance in the training phase and could generalize with small data sizes and a little loss of performance compared with an implementation that occupies a large dataset as the BiLSTM algorithm trained with the Physionet dataset. On the other hand, the LSTM approach has a similar performance to the ones that could be found in the literature with an accuracy of 91.25%, but in this case, the algorithm used all the information available to be trained, not only a small dataset remembering that this type of algorithm is greedier and needs to be trained with more information. In the computational time section, the LSTM needs less time to be trained from scratch than the time required to re-train the SquezeeNet. Also, the sensibility metric demonstrated with an 83.33% that both NN generalizes the information well.

An important factor for the election of an algorithm designed to achieve certain tasks (like MI classification in the present) is the available information for the training and testing phases. The quality and quantity of this information are elements that directly impact the computational time of the algorithm. This is because using a high-dimensional dataset obliges an improvement of the same through dimension reduction to achieve a good classification with a short computational time. On the other hand, if the data augmentation techniques are insufficient to achieve a dataset idoneous for the training phase with the information available, a high computational cost could be a good approach to prevent a performance loss.

REFERENCES

[1] G. Y. Wiederschain, "Biomedical engineering principles," *Biochemistry*, vol. 71, no. 5, p. 581, 2006.

[2] S. Madihally, *Principles of Biomedical Engineering*. Norwood, MA, USA: Artech House, 2019.

[3] M. X. Cohen, *Analyzing Neural Time Series Data: Theory and Practice*. Cambridge, MA, USA: MIT Press, 2014.

[4] A. B. Ritter, V. Hazelwood, A. Valdevit, and A. N. Ascione, *Biomedical Engineering Principles*. Boca Raton, FL, USA: CRC Press, 2011.

[5] N. Padfield, J. Zabalza, H. Zhao, V. Masero, and J. Ren, "EEG-based brain-computer interfaces using motor-imagery: Techniques and challenges," *Sensors*, vol. 19, no. 6, p. 1423, Mar. 2019.

[6] L. M. Ribas, F. T. Rocha, N. R. S. Ortega, A. F. da Rocha, and E. Massad, "Brain activity and medical diagnosis: An EEG study," *BMC Neurosci.*, vol. 14, no. 1, pp. 1–15, Dec. 2013.

[7] *World Report on Disability (ISBN 978 92 4 068521 5)*, World Health Organization, Geneva, Switzerland, vol. 30, 2011, p. 2015.

[8] K. Iwatsuki, M. Hoshiyama, S. Oyama, H. Yoneda, S. Shimoda, and H. Hirata, "Electroencephalographic functional connectivity with the tacit learning system prosthetic hand: A case series using motor imagery," *Frontiers Synaptic Neurosci.*, vol. 12, p. 7, Feb. 2020.

[9] J. J. Carr and J. M. Brown, *Introduction to Biomedical Equipment Technology*. Upper Saddle River, NJ, USA: Prentice-Hall, 2001.

[10] M. Al-Quraishi, I. Elamvazuthi, S. Daud, S. Parasuraman, and A. Borboni, "EEG-based control for upper and lower limb exoskeletons and prostheses: A systematic review," *Sensors*, vol. 18, no. 10, p. 3342, Oct. 2018.

[11] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016. [Online]. Available: http://www.deeplearningbook.org

[12] P. P. Cruz, *Inteligencia Artificial Con Aplicaciones a La Ingeniería*. Alfaomega, 2011.

[13] E. J. Rechy-Ramirez and H. Hu, "Bio-signal based control in assistive robots: A survey," *Digit. Commun. Netw.*, vol. 1, no. 2, pp. 85–101, Apr. 2015.

[14] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Ann. Eugenics*, vol. 7, no. 2, pp. 179–188, Sep. 1936.

[15] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "A training algorithm for optimal margin classifiers," in *Proc. 5th Annu. Workshop Comput. Learn. Theory*, Jul. 1992, pp. 144–152.

[16] J. Shuja, K. Bilal, W. Alasmary, H. Sinky, and E. Alanazi, "Applying machine learning techniques for caching in next-generation edge networks: A comprehensive survey," *J. Netw. Comput. Appl.*, vol. 181, May 2021, Art. no. 103005.

[17] S. Sotoudeh-Paima, A. Jodeiri, F. Hajizadeh, and H. Soltanian-Zadeh, "Multi-scale convolutional neural network for automated AMD classification using retinal OCT images," *Comput. Biol. Med.*, vol. 144, May 2022, Art. no. 105368.

[18] A. Cossu, A. Carta, V. Lomonaco, and D. Bacciu, "Continual learning for recurrent neural networks: An empirical evaluation," *Neural Netw.*, vol. 143, pp. 607–627, Nov. 2021.

[19] G. Zhang, V. Davoodnia, A. Sepas-Moghaddam, Y. Zhang, and A. Etemad, "Classification of hand movements from EEG using a deep attention-based LSTM network," *IEEE Sensors J.*, vol. 20, no. 6, pp. 3113–3122, Mar. 2020.

[20] D. Sarkar, R. Bali, and T. Ghosh, *Hands-On Transfer Learning With Python: Implement advanced Deep Learning and Neural Network Models Using TensorFlow and Keras*. Birmingham, U.K.: Packt, 2018.

[21] S. M. Shafiul Hasan, M. R. Siddiquee, R. Atri, R. Ramon, J. S. Marquez, and O. Bai, "Prediction of gait intention from pre-movement EEG signals: A feasibility study," *J. NeuroEng. Rehabil.*, vol. 17, no. 1, pp. 1–16, Dec. 2020.

[22] X. Deng, B. Zhang, N. Yu, K. Liu, and K. Sun, "Advanced TSGL-EEGNet for motor imagery EEG-based brain-computer interfaces," *IEEE Access*, vol. 9, pp. 25118–25130, 2021.

[23] V. J. Lawhern, A. J. Solon, N. R. Waytowich, S. M. Gordon, C. P. Hung, and B. J. Lance, "EEGNet: A compact convolutional neural network for EEG-based brain-computer interfaces," *J. Neural Eng.*, vol. 15, no. 5, Oct. 2018, Art. no. 056013.

[24] Z. Khademi, F. Ebrahimi, and H. M. Kordy, "A transfer learning-based CNN and LSTM hybrid deep learning model to classify motor imagery EEG signals," *Comput. Biol. Med.*, vol. 143, Apr. 2022, Art. no. 105288.

[25] W. Huang, W. Chang, G. Yan, Z. Yang, H. Luo, and H. Pei, "EEG-based motor imagery classification using convolutional neural networks with local reparameterization trick," *Exp. Syst. Appl.*, vol. 187, Jan. 2022, Art. no. 115968.

[26] G. Schalk, D. J. McFarland, T. Hinterberger, N. Birbaumer, and J. R. Wolpaw, "BCI2000: A general-purpose brain-computer interface (BCI) system," *IEEE Trans. Biomed. Eng.*, vol. 51, no. 6, pp. 1034–1043, Jun. 2004.

[27] S. M. Abdelfattah, G. M. Abdelrahman, and M. Wang, "Augmenting the size of EEG datasets using generative adversarial networks," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2018, pp. 1–6.

[28] GA of the Wirld Medicla Association, "World medical association declaration of Helsinki: Ethical principles form medical research involving human subjects," *J. Amer. College Dentist*, vol. 81, no. 3, pp. 14–18, 2014.

[29] P. Nagabushanam, S. Thomas George, and S. Radha, "EEG signal classification using LSTM and improved neural network algorithms," *Soft Comput.*, vol. 24, no. 13, pp. 9981–10003, Jul. 2020.

[30] E. Netzer, A. Frid, and D. Feldman, "Real-time EEG classification via coresets for BCI applications," *Eng. Appl. Artif. Intell.*, vol. 89, Mar. 2020, Art. no. 103455.

[31] H. U. Amin, W. Mumtaz, A. R. Subhani, M. N. M. Saad, and A. S. Malik, "Classification of EEG signals based on pattern recognition approach," *Frontiers Comput. Neurosci.*, vol. 11, p. 103, Nov. 2017.

[32] M. K. Ghumman, S. Singh, N. Singh, and B. Jindal, "Optimization of parameters for improving the performance of EEG-based BCI system," *J. Reliable Intell. Environ.*, vol. 7, no. 2, pp. 145–156, Jun. 2021.

[33] A. Al-Saegh, S. A. Dawwd, and J. M. Abdul-Jabbar, "Deep learning for motor imagery EEG-based classification: A review," *Biomed. Signal Process. Control*, vol. 63, Jan. 2021, Art. no. 102172.

[34] F. Karimi, J. Kofman, N. Mrachacz-Kersting, D. Farina, and N. Jiang, "Detection of movement related cortical potentials from EEG using constrained ICA for brain-computer interface applications," *Frontiers Neurosci.*, vol. 11, p. 356, Jun. 2017.

[35] M. H. Alomari, A. Samaha, and K. AlKamha, "Automated classification of L/R hand movement EEG signals using advanced feature extraction and machine learning," 2013, *arXiv:1312.2877*.

[36] S. M. Khan, A. A. Khan, and O. Farooq, "Selection of features and classifiers for EMG-EEG-based upper limb assistive devices—A review," *IEEE Rev. Biomed. Eng.*, vol. 13, pp. 248–260, 2020.

[37] H. Misra, S. Ikbal, H. Bourlard, and H. Hermansky, "Spectral entropy based feature for robust ASR," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, May 2004, p. 193.

[38] E. Da Silva and D. Sampson, "Successive approximation wavelet vector quantization for image and video coding," in *Advances in Imaging and Electron Physics*. vol. 97. Alfaomega, 1996, pp. 191–255.

[39] D. J. Ewins, S. S. Rao, and S. G. Braun, *Encyclopedia of Vibration, Three-Volume Set*. Cambridge, MA, USA: Academic Press, 2002.

[40] B. Boashash, *Time-Frequency Signal Analysis and Processing: A Comprehensive Reference*. Cambridge, MA, USA: Academic Press, 2015.

[41] R. Bousseta, I. El Ouakouak, M. Gharbi, and F. Regragui, "EEG based brain computer interface for controlling a robot arm movement through thought," *IRBM*, vol. 39, no. 2, pp. 129–135, Apr. 2018.

[42] L. Hu and Z. Zhang, *EEG Signal Processing and Feature Extraction*. Berlin, Germany: Springer, 2019.

[43] C. Ding and H. Peng, "Minimum redundancy feature selection from microarray gene expression data," *J. Bioinf. Comput. Biol.*, vol. 3, no. 2, pp. 185–205, Apr. 2005.

[44] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.

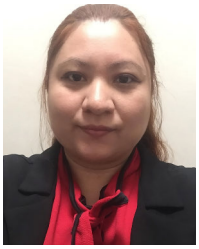
[45] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and <0.5 MB model size," 2016, *arXiv:1602.07360*.

[46] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017.

[47] H. Sugata, M. Hirata, T. Yanagisawa, K. Matsushita, S. Yorifuji, and T. Yoshimine, "Common neural correlates of real and imagined movements contributing to the performance of brain-machine interfaces," *Sci. Rep.*, vol. 6, no. 1, pp. 1–11, Apr. 2016.

[48] M. C. Ng, J. Jing, and M. B. Westover, *Atlas of Intensive Care Quantitative EEG*. Berlin, Germany: Springer, 2019.

[49] M. Li, M. Zhang, X. Luo, and J. Yang, "Combined long short-term memory based network employing wavelet coefficients for MI-EEG recognition," in *Proc. IEEE Int. Conf. Mechatronics Autom.*, Aug. 2016, pp. 1971–1976.
[50] Z. Wan, R. Yang, M. Huang, N. Zeng, and X. Liu, "A review on transfer learning in EEG signal analysis," *Neurocomputing*, vol. 421, pp. 1–14, Jan. 2021.
[51] K. Zhang, N. Robinson, S.-W. Lee, and C. Guan, "Adaptive transfer learning for EEG motor imagery classification with deep convolutional neural network," *Neural Netw.*, vol. 136, pp. 1–10, Apr. 2021.
[52] T. Radüntz, "Signal quality evaluation of emerging EEG devices," *Frontiers Physiol.*, vol. 9, p. 98, Feb. 2018.
[53] A. S. Oliveira, B. R. Schlink, W. D. Hairston, P. König, and D. P. Ferris, "Proposing metrics for benchmarking novel EEG technologies towards real-world measurements," *Frontiers Hum. Neurosci.*, vol. 10, p. 188, May 2016.



ALICIA GUADALUPE LAZCANO-HERRERA was born in Mexico City, in 1991. She received the B.S. degree in mechatronic engineering and the M.Sc. degree in engineering from Instituto Tecnológico de Tlalnepantla (ITTLA), in 2016 and 2019, respectively. She is currently pursuing the Ph.D. degree in computer sciences with Tecnologico de Monterrey. From 2017 to 2020, she imparted classes in the science, technology, engineering, and mathematics (STEM) field at diverse educational levels and schools, being the last at Universidad Tecnológica de México (UNITEC). Since 2020, she has been a Teacher Assistant with the Sciences and Engineering School, Tecnologico de Monterrey. Her research interests include machine learning algorithms, signal processing, brain–computer interfaces, and rehabilitation applications.

RITA Q. FUENTES-AGUILAR received the higher studies degree (Hons.) in biomedical engineering from Instituto Politecnico Nacional (IPN) and the M.Sc. and Ph.D. degrees with a specialty in automatic control from the Center for Research and Advanced Studies (CINVESTAV), IPN. She is currently a Researcher with the Institute of Advanced Materials for Sustainable Manufacturing, Tecnológico de Monterrey, and the Leader of the Research Group with a focus on enabling technologies. She holds three patents, seven national patent applications, three PCTs, as well as four public copyright registries. Her research interests include mathematical model identification and control, biomedical applications, machine learning for prediagnosis, and the development of solutions in the medical area. She is a member of the National System of Researchers.

ADRIAN RAMIREZ-MORALES was born in Guanajuato, Mexico, in 1985. He received the B.S. degree in bionic engineering from the Interdisciplinary Professional Unit in Engineering and Advanced Technologies, Instituto Politecnico Nacional (IPN), Mexico, in 1998, and the M.S. and Ph.D. degrees in cellular and molecular neurobiology from the Center for Research and Advanced Studies, IPN, in 2011 and 2019, respectively. Since 2010, he has been a Professor with the bionic and biomedical engineering academies in higher education centers, such as IPN, La Salle University, and the Institute of Technology and Higher Studies of Monterrey. His research interests include the study of central pattern generators in the central nervous system through computational models, the implementation of brain-machine interfaces for medical use, and the design and construction of biomimetic devices.

MARIEL ALFARO-PONCE (Member, IEEE) received the bachelor's degree in biomedical engineering, the Master of Science degree in microelectronic engineering, and the Ph.D. degree in computer science from Instituto Politecnico Nacional, Mexico. She has been a member of the National System of Researchers of Mexico (SNI-Level I). Since 2022, she has been the Head of the Manufacturing Processes for Advanced Materials CDMX Research Unit. She is currently an Assistant Professor with the Biomedical Engineering Program, Tecnologico de Monterrey Ciudad de Mexico. Her research interests include artificial intelligence, rehabilitation devices, and intelligent bioinstrumentation.