Biomedical Signal Processing and Control 68 (2021) 102752

Contents lists available at ScienceDirect

Biomedical Signal Processing and Control

journal homepage: www.elsevier.com/locate/bspc

Deep learning for surface electromyography artifact contamination type detection

Juliano Machado [a,*], Amauri Machado [b], Alexandre Balbinot [a]

[a] Laboratory of Electro-Electronic Instrumentation (IEE-Bio), Post-graduate Program in Electrical Engineering (PPGEE), Department of Electrical Engineering, Federal University of Rio Grande do Sul (UFRGS), Avenue Osvaldo Aranha 103, 206-D, Porto Alegre, RS, Brazil
[b] Department of Statistics and Informatic, Physics and Mathematic Institute, Federal University of Pelotas (UFPEL), Postal Box #354 - 96001-970, Pelotas, RS, Brazil

ARTICLE INFO

Keywords:
Surface electromyography
Contaminants
Quality
Recurrent neural network
Long short-term memory

ABSTRACT

The quality of surface Electromyography (sEMG) signals could be an issue if highly contaminated by Power Line Interference (PLI), Electrocardiogram signal (ECG), Movement Artifact (MOA) or White Gaussian Noise (WGN), that could lead to unsafe operation of devices that is controlled by sEMG data, such as electro-mechanical prothesis. There are some mitigation methods proposed for some specifics sEMG contaminants and to use these methods in an efficient way is important to identify the contaminant in the sEMG signal. In this work we propose the use of a Recurrent Neural Network (RNN) using Long Short-Term Memory (LSTM) units in the hidden layer with no need of features extraction with the objective to classify the signal directly from sequences of the band-pass filtered data. The method proposed use the NinaPro database with amputee and non-amputee subjects. Only non-amputee subjects are used for parameters selection and then tested on both databases. The results show that 98% of the non-contaminated sEMG data was corrected classified and more than 95% of the contaminants were identified inside the training SNR range. Also, in this work is presented a SNR sensibility control and the contamination analysis in the range from −40 dB to 40 dB in 10 dB steps. The conclusion is that is possible to classify the contamination type in sEMG signals with a RNN-LSTM with a 112.5 ms time window and to predicted with a small error the classification hit rate for each SNR level in some cases.

1. Introduction

The surface electromyography (sEMG) signal has applications that could be clinic [1,2], in rehabilitation [3,4] or even in wearables [5,6]. The quality of the signal could be an issue for practical use in these applications if it is highly contaminated. There are many contaminants types that could lead to a misinterpreted result by a human operator or Computational Intelligence (CI) system for Pattern Recognition (PR).

These contaminants could be Electrocardiogram (ECG) signal, Power Line Interference (PLI), Movement Artifact (MOA) and additive White Gaussian Noise (WGN) that could represent thermal noise or a loose electrode. There are others type of contaminants, but the ones cited previously are presented in almost every paper dealing with sEMG quality analysis [7–16].

Not every action is the same to mitigate the contamination, and the knowledge of the contamination type could be useful. For example, in [17] is presented a method to remove ECG contamination from sEMG signals with an adaptive neuro-fuzzy inference system (ANFIS) and wavelets transforms. This method could have more time-processing efficiency if triggered only when the sEMG is contaminated by ECG.

Even simple actions for mitigation, like a 20 Hz corner filter for MOA mitigation [9,18] or a notch filter for PLI mitigation could be very costly if the number of channels increases. For electrode displacement, that has similar behavior as an additive WGN [12,14,15], could be very useful in clinical applications, alerting the operator to replace the electrode.

Our objective is limited to a main contaminant in the data, that is possible to occur, as in Targeted Muscle Reinnervation (TMR) when the procedure reinnervate in the chest muscle, that is highly contaminated with ECG [19] or MOA in wearables devices. The methodology presented in this paper is a basis for future analysis of contaminants mixtures, where a model will have to deal with multiples output classes containing the contaminants and its possible mixtures.

In [7] the authors made studies in the diaphragm muscles with invasive EMG signals and developed a quadratic regression model that uses four features being that two are in time-domain and the others in frequency domain. The contaminants used in this study were ECG and

* Corresponding author.
E-mail addresses: julianoc.machado@gmail.com (J. Machado), amachado@ufpel.edu.br (A. Machado), abalbinot@gmail.com (A. Balbinot).

https://doi.org/10.1016/j.bspc.2021.102752
Received 26 August 2020; Received in revised form 4 April 2021; Accepted 9 May 2021
Available online 24 May 2021

Table 1
Summarization on sEMG contaminant identification papers.

Paper	Contaminats	SNR Levels	# Features	Model
[7]	ECG; MOA; WGN	Not specified	4	Quadratic Regressor
[8]	ECG; MOA; WGN	Not specified	4	Features index
[10]	PLI; MOA; ECG; A/D *clipping;* quantization error; saturation	−20 dB to 20 dB	10	SVM
[12]	MOA, PLI, ECG, Saturation and WGN	−20 dB to 20 dB	6	SVM *one-vs-one* for 5 classes
[13]	MOA; Displaced Electrode; WGN	Not specified	4	Mahalanobis outlier detector
[15]	MOA; Saturation; WGN; PLI; ECG	Not specified	5	SVM
[16]	WGN; ECG; PLI; MOA	Not specified	12	SOM

MOA, and no information about the contaminant type where given.

Other study that uses the diaphragm EMG signal and ECG contamination is [8] and the objective were to identify the degree of contamination with four indexes, all in frequency domain. In [9], the objective was to mitigated the contamination by ECG and baseline noise, such as MOA and a high pass corner filter with cutoff frequency from 20 Hz was proposed for applications involving isometric contractions or natural and common movements.

There are some works that uses outliers identification [10,11,13], with the main advantage that previous information about the contamination is not needed. Statistical tools or CI is used for outliers identification, such as normality test [11], Mahalanobis Distance [13] and an one class Support Vector Machine (SVM) [10]. None of the outlier detectors could identify the contamination type and all of them uses extracted features from the sEMG signal.

The SVM was used in [15] with binary classification separating the clean signal and signal contaminated, without SNR control, with ECG, electrode displacement, simulated as an additive WGN, PLI, MOA and Saturation using five time domain features

In [12] the SVM is also used, but in this case it was used as a multi class classifier with a one vs one configuration. The SVM did not differentiate the clean signal from the contaminated, it only classify the contamination type in different SNR levels. For a whole automatic system, it is needed a previous quality analysis, such as the outlier detectors. ECG, PLI, MOA, WGN and Saturation were used as contaminants for detection and seven features were used.

Self-Organizing Maps (SOM) are a unsupervised machine learning method that is used by [16] to identify contamination type in sEMG data from 4 channels in the back muscles and synthetic data. The baseline noise, artificially generated MOA, ECG interference, PLI and a mixture of anomalies were used.

The SOM method was designed using wavelet decomposition coefficients and features selection by a robust Principal Component Analysis (rPCA). The perform for was measured by recall and precision, obtaining a recall above 95%, except MOA contamination, and precision above 80%. No SNR range was reported and for an Intel i7 CPU it took 25 ms to run the algorithm. In Table 1, is summarized the principal publications about contaminant identification on sEMG data.

The use in real time is essential for robotic based rehabilitation systems, avoiding any unsafe operation that could occur from the processing of the contaminated sEMG signal [14]. A way to improve the time efficiency is to reduce processing steps, such as feature extraction.

One way to make quality analysis in sEMG signal without feature extraction is using sequence prediction tools such as the Recurrent Neural Networks (RNN). Considering that the sEMG signal and its mixture with contaminants in different SNR levels will have different sequences in a time window, we make the hypothesis that it is possible to differentiate contaminated from non-contaminated signal and, if contaminated, the main contamination type using a sequence window of the sEMG data.

This paper is organized as follows: in Section 2 the methodology is presented; Section 3 evaluates and analyses the results; the discussion is presented in Section 4 and the conclusions are made in Section 5.

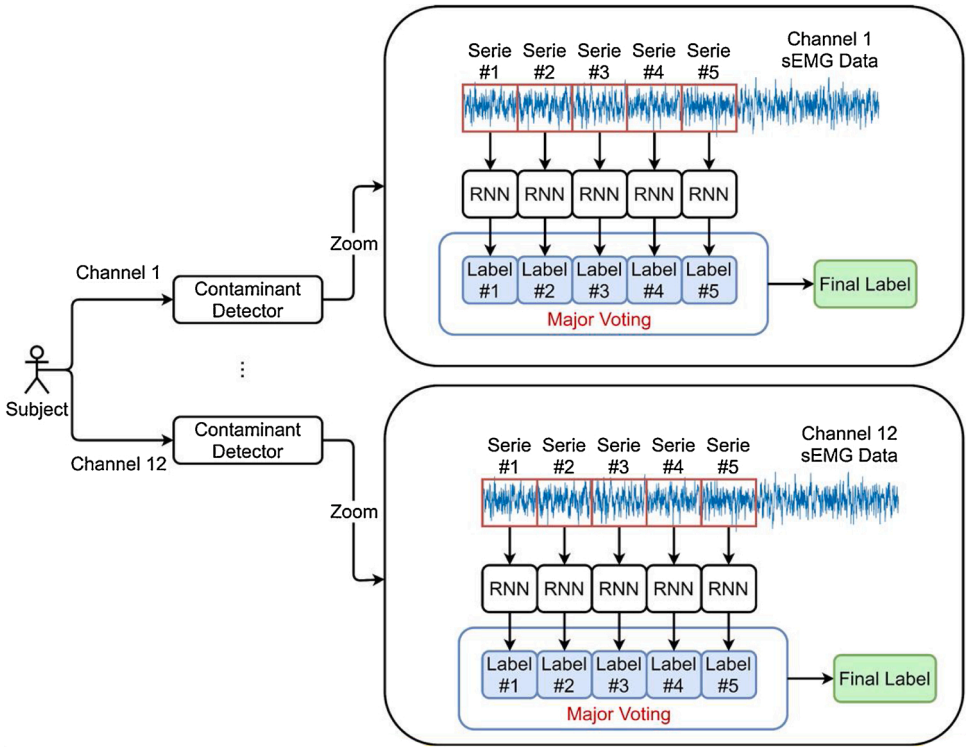

Fig. 1. Proposed pipeline for sEMG contamination detection.

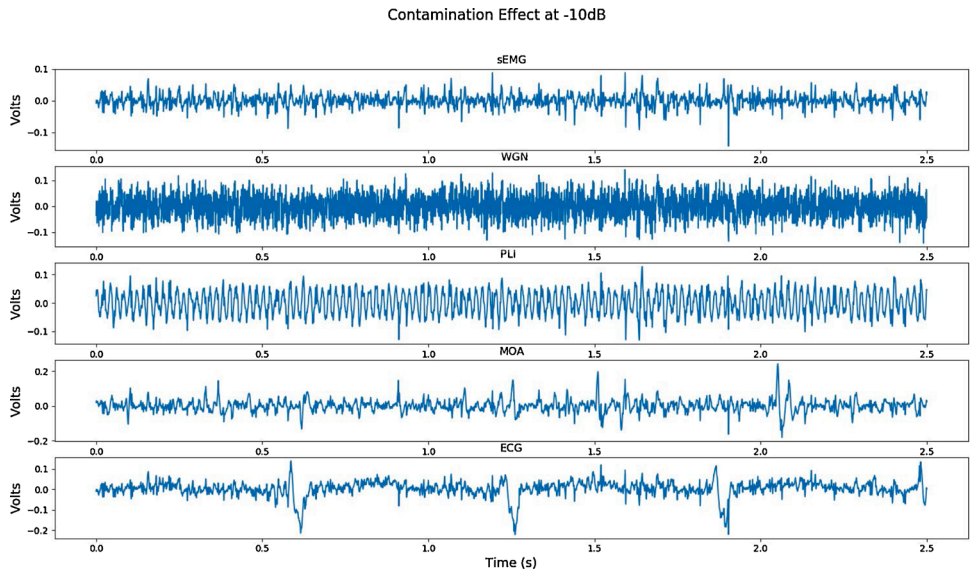

Fig. 2. Contamination effect at SNR = −10 dB.

2. Methodology

In Fig. 1 is presented the proposed processing pipeline for all sEMG channels. One RNN model is used in each channel and classify a N size sequence. The label is assigned using major voting in the last 5 classified sequences without overlapping. There are five classes to be assigned to Final Label: Clean, WGN, PLI, MOA and ECG. All algorithms were made in Python 3.7.7 using mainly the following libraries: GPU tensorflow, numpy and scipy.

The results are presented for three main objectives:

1. Model Selection: A methodology based in the standard deviation is used to define the series length for each RNN input and the channel used to train a general model. A statistical analysis to define the LSTM hidden units is applied as well. More details about the model architecture is provided in Section II.
2. Model Testing: The efficiency of the model is tested for nine SNR levels, using non-amputee and amputee individuals. The metric used is the hit rate accuracy.
3. SNR Sensitive: Considering that the expect behavior of the curve SNR x hit rate is given by the non-amputee testing data.

2.1. NINAPro sEMG database

The publicly available NINAPro (Non-Invasive Adaptive Prosthetics) database of sEMG signals was used in this study. The NINAPro database uses 12 active wireless electrodes (Delsys TM Trigno Wireless System®). NINAPro data is acquired through National Instruments® NI-DAQ PCMCIA 6024E platform at a rate of 2 kHz, 12 bits and with a noise lower than 750 nV RMS [4,20]. It is important to highlight that this study does not use any feedback system and does not monitor the force neither the way the volunteer performs the movements.

The sEMG database of each subject consists of one session with 17 movements for exercise 1 and 22 movements for exercise 2. Each one of the distinct movements is repeated 6 consecutive times, during 5 s each movement, separated by 3 s rest time. The sEMG data is acquired through 12 surface electrodes. The twelve electrodes are divided into eight electrodes uniformly spaced just beneath the elbow at a fixed distance from the radio-humeral joint, two electrodes on the flexor digitorum and the extensor digitorum, and two electrodes on the main activity spots of the biceps and the triceps.

There are 40 intact-upper-member volunteers, consisting of 28 men and 12 women, 34 right-handed and 6 left-handed with 29.9 ± 3.9 y mean age. Data from another 11 volunteers with some level of amputation in the upper members from the same database were used [21]. A postprocessing stage was performed with a Hampel filter to remove 50 Hz interference.

2.2. Artifact contamination

To remove or attenuate low and high frequency contamination that could be present, all data were filtered between 20 and 500 Hz with a fourth order Butterworth, normalized by the maximum absolute value of each channel and labeled as non-contaminated sEMG data.

A simple band-pass filtering will not remove all possible artifacts, but in studies that uses this database for movement type recognition in the non-amputee subjects [22] achieved overall accuracy (movement type and rest period) of 94.4% ± 7.6% using a regularized version of an Extreme Learning Machine (R-ELM). This is an indicative that no significant contamination affects the data.

For other side, using the same R-ELM with the NinaPro amputee database, an overall accuracy of 76.3% ± 18.2% was achieved, indicate a less integrity in the data that could be caused by artifact contamination or a more difficult characterization of the sEMG data from an amputated muscle.

Before contaminating the data, a method to control the SNR level was defined. Previous research showed no standard way on how to determine the signal SNR for contamination. The Consensus for Experimental Design in Electromyography (CEDE) project [23] defines the SNR as the amplitude of the EMG signal relative to the recording noise.

The SNR method used by [12] depends on fractions of the Maximum Voluntary Contraction (MVC) that were measure for the five analyzed subjects in the study. In NINAPro database, no MVC information was given and since different subjects will have different contraction levels, in order to make a standard way for all individual, the mean power from each subject was calculated by Eq. (1), where x_n is the n-th sample from the N size rest samples from one individual. Fig. 2 shows the contamination effect at SNR = −10 dB for a 2.5 s rest segment.

$$Rest\ Power\ Subject = \frac{1}{N}\sum_{N} x_n^2 \tag{1}$$

2.2.1. Movement artifact

An experimental setup, with 12 channels disposed the same way as in

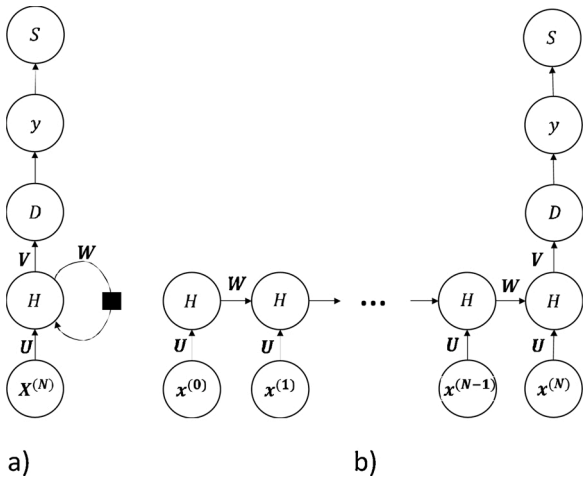

Fig. 3. RNN architecture. a) folded b) unfolded.

the NinaPro protocol, was built. It was used 24 disposable surface electrodes and a reference electrode, placed on the individual's forehead. The channels were connected to a battery-powered commercial sEMG device (EMG 830 C, from EMG System do Brasil). The signal was digitalized at a 2-kHz sample frequency and 18 bits of quantization by a NI USB-6289 platform from National Instruments. The acquisition was performed using a notebook running LabVIEW 9 on a Windows 10 environment.

With no contraction, a sequence of 10 taps with 1 s interval were performed from one individual, age of 37y, 183 cm height and 80 kg weight to physically dislocate the electrode from the muscle to produce the MOA. A similar methodology was used in [9]. This was repeated for each channel. A moving-average filter with 11 samples was used to smooth the data.

The Clean Data for each channel is added with a vector containing randomly selected MOA from the same channel. The MOA power is calculated with same method from Eq. (1) and then the data is multiplied by a factor to meet the desired SNR.

2.2.2. Electrocardiogram artifact

The ECG data was acquired in the publicity available Physionet data bank [24]. Segments of 10 s were randomly selected from one channel for 17 individuals. An up-sample procedure was necessary to fit sEMG data sample rate.

Then a vector with the 10 s segments was multiplied by a factor to meet the desire SNR and added to the Clean Data. The ECG power is calculated with the same method from Eq. (1).

2.2.3. Power line interference

A 50 Hz wave, with power given by Eq. (2) is added to the Clean Signal. The amplitude A is multiplied by a factor to meet the desired SNR.

$$P_{Power\ Line} = \frac{A^2}{2} \tag{2}$$

2.2.4. Additive White Gaussian Noise

A normally distributed random vector with zero mean is added. The variance is adjusted to meet the desired SNR.

2.3. Recurrent neural network

The RNN is a network with memory specialized in time series classification as speech [25], forecasting [26], fault detection [27] and others applications where one label is assigned for a sequence of data points. The proposed architecture is detailed in Fig. 3 and the memory element is present in the H layer where the information passes through time by the $\boldsymbol{W}$ set of weights. The input layer passes a sequence $\boldsymbol{X}^{(N)} \in \mathbb{R}^{F\times N}$ by the $\boldsymbol{U}$ set of weights, where F is the number of features, N is the sequence total length and $\boldsymbol{x}^{(n)}$ is the n-th sample with F features from the input sequence.

The D layer represents a Dropout layer. In this layer, that is a fully connect layer by the $\boldsymbol{V}$ set of weights, each unit has a probability p that the set of weights will be trained per epoch. This improves the generalization capability of the network [28]. The S layer applies a *softmax* function in the output y and the class is selected by applying the *argmax* function in the *softmax* output.

The output from layer H is given by Eqs. (3) and (4), where a_h^n is the output from unit h in time sample n, considering F features and H' recurrent connections. u_{fh} and $w_{h'h}$ are the input and recurrent weights, respectively. The hyperbolic tangent (*tanh*) activation function $\theta(\cdot)$ gives the h unit its final output in time sample n, b_h^n.

$$a_h^n = \sum_{f=1}^{F} u_{fh} x_f^n + \sum_{h'=1}^{H'} w_{h'h} b_{h'}^{n-1} \tag{3}$$

$$b_h^n = \theta\left(a_h^n\right) \tag{4}$$

The memory units used in layer H were the Long Short-Term Memory (LSTM) type in order to avoid known issues from RNN that is the exploding and/or the vanishing gradient effect [29,30]. There are variants of the LSTM, but [31] show that none of them improved the standard LSTM-RNN.

The LSTM unit has 3 gates that is used to control the information flow named *Input Gate*, *Output Gate* and *Forget Gate*. The *Input Gate* and the *Output Gate* determines where an input or output from a unit is relevant for the network output label. The *Forget Gate* determines where an information from a previous time step is relevant for the network label output.

The Eq. (3) must be transformed in a set of equation, to handle the new parameters added to the model by the control gates. The Eqs. (5)–(9) for the forget, input and output gates and input layer, respectively, gives the b_h^n LSTM unit output. $\theta_s(\cdot)$ and $\theta_t(\cdot)$ are the sigmoid activation function and the hyperbolic tangent activation function.

$$Fg_\varphi^n = \theta_s\left(\sum_{f=1}^{F} u_{f\varphi} x_f^n + \sum_{h=1}^{H'} w_{h\varphi} b_h^{n-1}\right) \tag{5}$$

$$Ig_i^n = \theta_s\left(\sum_{f=1}^{F} u_{fi} x_f^n + \sum_{h=1}^{H'} w_{hi} b_h^{n-1}\right) \tag{6}$$

$$Og_o^n = \theta_s\left(\sum_{f=1}^{F} u_{fo} x_f^n + \sum_{h=1}^{H'} w_{ho} b_h^{n-1}\right) \tag{7}$$

$$IL_\gamma^n = \theta_t\left(\sum_{f=1}^{F} u_{f\gamma} x_f^n + \sum_{h=1}^{H'} w_{h\gamma} b_h^{n-1}\right) \tag{8}$$

$$b_h^n = \theta_t\left(c^{n-1} \odot Fg_\varphi^n + Ig_i^n \odot IL_\gamma^n\right) \odot Og_o^n \tag{9}$$

In the Eq. (9) is clear that the forget gate controls the importance of the last memory cell state and the input gate controls the importance from the actual input. At last, the output gate defines the output importance, based on the input and past output.

After all N data points were presented to the network, the data passes to a dropout dense layer, with dropout rate equals to 0.5, providing the output $s = softmax(y)$ and an $argmax(s)$ gives the final class.

There are many parameters that could be tuned in the LSTM-RNN proposed, such as the p probability in the Dropout layer, number of LSTM units in the H layer, the N sequence size, the mini-batch training size, and the number of epochs, among others concerning the architecture.

Table 2
LSTM-RNN Parameters Settings.

Dropout rate	0.5 for all configurations
LSTM Units	50, 75 and 150
Sequence Length (*N*)	15, 30, 45 and 60
Epochs	20 for all configurations
Mini-batch size	128 for all configurations
Features (*F*)	1 for all configurations

To limit the scope of analysis in this paper, only the LSTM and the *N* sequence length will be tested in 3 levels each, as Table 2 shows.

2.4. Training and testing data

Considering that the PLI contaminant data is periodic, the ECG is quasi-periodic as the MOA during gait, for example, and is composed by low frequencies components [32], a small dataset could be sufficient for data generalization with 9 individuals randomly chosen from the non-amputee database. As WGN has a spectral power different from all other classes, we suppose that the model will differentiate all classes even with a small dataset, compared as the test dataset.

From the 9 individuals of the non-amputee exercise 1 database, 12 models were trained from each channel using 3 SNR levels according to Fig. 4. To be clearer, the model #7, for example, is trained using the clean data from the channel 7 from all 9 individuals. Groups of 3 individuals will have the channel 7 data contaminated with 3 different SNR levels.

For the SNR control, it is used three different training set to test the data with different SNR ranges, according Table 2. For optimal model parameters selection presented in Table 1 (LSTM units and Sequence Length), only the Training Data C, described in Table 3 is used. The Training Data A and B are used to test the SNR-sensitive control feature.

To format the data to feed the RNN input, a 3-dimensional array was used, consisting in [samples, time steps (*N*), features (*F*)]. The one-dimensional vector with $F = 1$ feature for each class was split in k samples of time series, that belong to one of the five class, with *N* steps in each sample, generating one feature and a five-class data array with $5*k$

Table 3
SNR Training Levels.

	SNR Level 1	SNR Level 2	SNR Level 3
Training Data A	−20 dB	−10 dB	0 dB
Training Data B	−30 dB	−20 dB	−10 dB
Training Data C	−40 dB	−30 dB	−20 dB

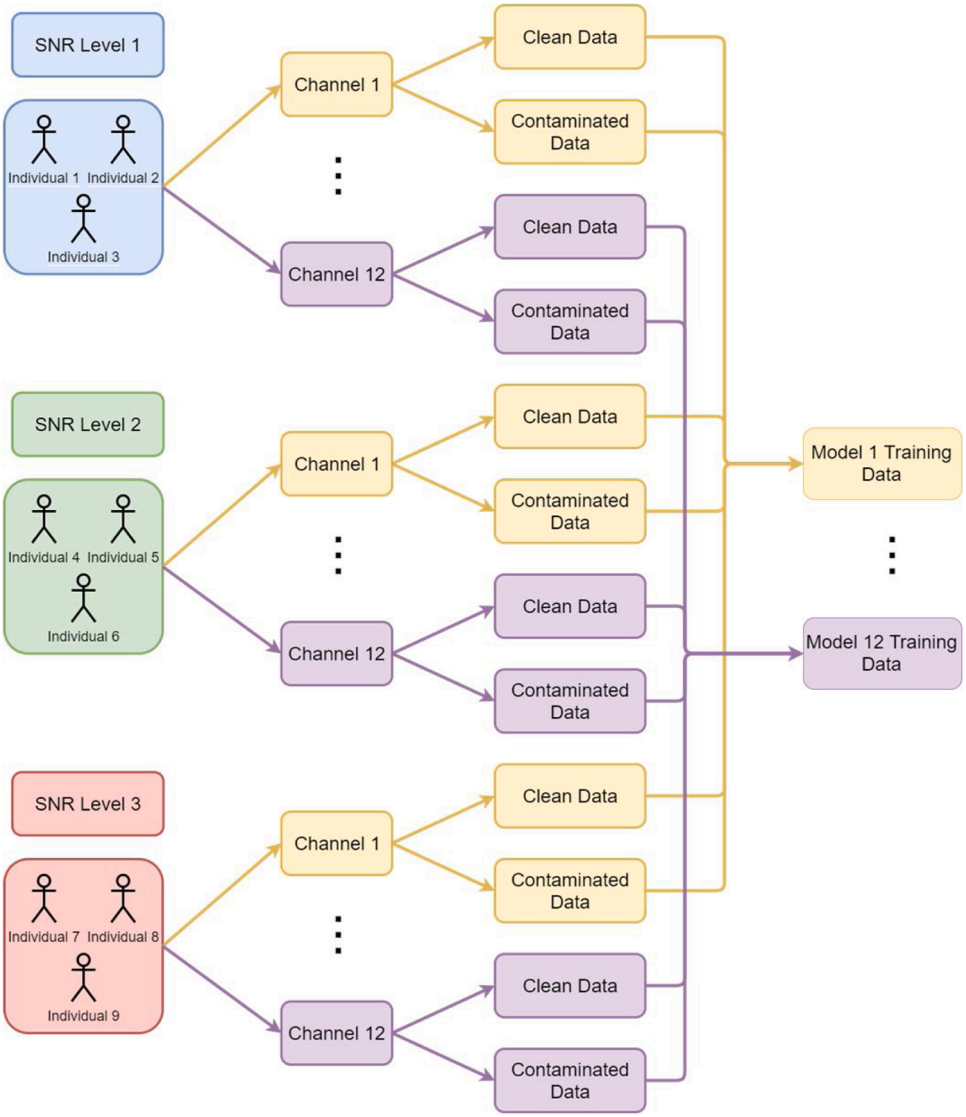

Fig. 4. Generating training data with Clean Data and different SNR Levels.

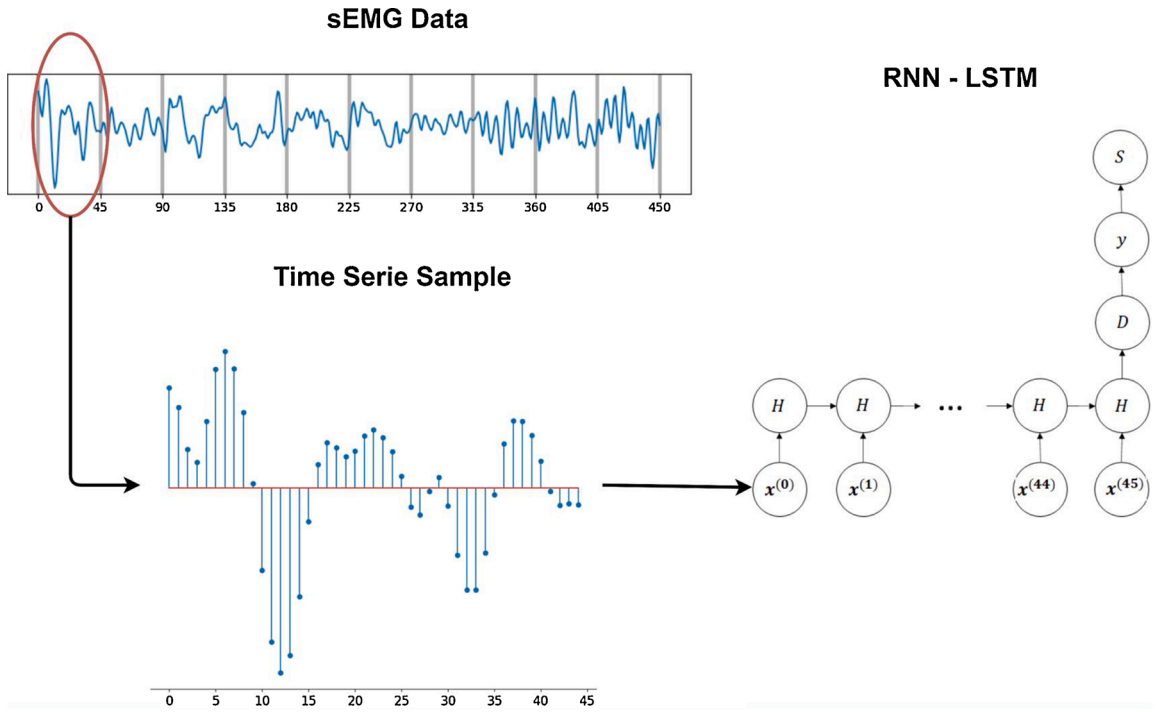

Fig. 5. Example of sEMG data segmentation from one channel to input the RNN-LSTM. In this example, N = 45 and 10 series samples are presented (k = 10).

samples. In Fig. 5 is presented an example of data segmentation from one channel, using $N = 45$ with $k = 10$ time series samples, i.e. the classification will result in 2 class label from the major voting process (one for each *5* time series samples). The method used to train was the Back-Propagation Through Time (BPTT) [33] using the Adam optimizer [34].

The testing data were composed by clean and contaminated data from the 31 volunteers that were not used for training (exercise 1) and the 11 amputee volunteers (exercise 1), split in sequences with the same process used for the training data. The contamination is made equally for all the test volunteers in 9 SNR levels ranging from -40 dB to 40 dB in 10 dB steps. All models were used to classify all volunteers' channels.

In time series classification, the sequence length N is a critical parameter, as the number of LSTM hidden units for LSTM-RNN. To choose a well-balanced architecture, the sequence length N from Table 2 was selected with an analysis on the lower standard deviation (SD) of the classification hit rate in the same SNR interval used for training. All individuals, channels, models, and classes in the non-amputee test database were used. The process is repeated for all number of LSTM units from Table 2.

With 12 channels to train, there are some options to train a model, such a model per-channel, but for sake of simplification only one channel is used to train the final model. Considering the selected N sequence length, the data from the classification of all 12 models in the Training Data C range are used for a Levene's variance homogeneity test with significance level equal 5% Considering that the models have at least two non-homogeneous variances, it means that has at least two different variances, and the model with the lower variance (or SD) will be selected.

With the sequence size and channel used for model training selected, a statistical analysis was performed in the classification hit rate for the different number of LSTM units and SNR intervals. It was used the General Linear Model (GLM) procedure of SAS/STUDIO® 5.2 statistical software for each combination of channel and class, considering each subject as an observation.

The LSTM-SNR Interval analysis were made between −40 dB and 10 dB because, a 20 dB or higher SNR is almost equal to the non-contaminated sEMG data, accordingly Table 4. The goal is to choose the model with the minimum possible LSTM units because each unit insert more parameters for the gates, impacting in time-processing and memory consuming.

After all parameters were selected, the model is tested for the amputee and non-amputee database with the three different training data shown in Table 3, to evaluate the sensibility. The results obtained for the non-amputee database exercise 1, the same used to train the model, is used as standard to predict the expected hit rate in each SNR point for each training set. The use of non-amputee subjects is thinking in a future real application, where is easier to find non-amputee subjects for data acquisition.

Table 4
Mean Autocorrelation for all channels from -40 dB to 40 dB.

		CHANNELS											
		Ch1	Ch2	Ch3	Ch4	Ch5	Ch6	Ch7	Ch8	Ch9	Ch10	Ch11	Ch12
SNR INTERVAL	−40 dB	0.24	0.04	0.03	0.06	0.02	0.02	0.08	0.08	0.03	0.06	0.01	0.01
	−30 dB	0.60	0.13	0.09	0.17	0.05	0.06	0.23	0.23	0.10	0.17	0.05	0.03
	−20 dB	0.91	0.36	0.26	0.46	0.14	0.15	0.53	0.56	0.30	0.46	0.14	0.08
	−10 dB	0.99	0.71	0.60	0.81	0.38	0.36	0.84	0.83	0.65	0.80	0.37	0.24
	0 dB	1.00	0.93	0.88	0.97	0.71	0.71	0.97	0.94	0.89	0.96	0.70	0.57
	10 dB	1.00	0.99	0.98	1.00	0.93	0.91	1.00	0.99	0.97	1.00	0.91	0.86
	20 dB	1.00	1.00	1.00	1.00	0.99	0.95	1.00	1.00	0.99	1.00	0.98	0.97
	30 dB	1.00	1.00	1.00	1.00	1.00	0.97	1.00	1.00	1.00	1.00	1.00	1.00
	40 dB	1.00	1.00	1.00	1.00	1.00	0.98	1.00	1.00	1.00	1.00	1.00	1.00

Table 5
Mean hit rate and Standard Deviation, in % of correct classified sEMG for a major voting for 5 segments.

		SNR Interval								
N	LSTM	−40 dB	−30 dB	−20 dB	−10 dB	0 dB	10 dB	20 dB	30 dB	40 dB
15	50	82.40 ± 37.11	85.48 ± 33.06	75.89 ± 34.57	46.68 ± 43.59	24.05 ± 41.15	20.05 ± 39.43	20.01 ± 39.44	20.01 ± 39.44	20.01 ± 39.44
	75	84.81 ± 35.13	86.03 ± 32.63	76.16 ± 34.45	47.53 ± 43.52	24.55 ± 41.42	20.06 ± 39.43	20.01 ± 39.44	20.01 ± 39.44	20.01 ± 39.44
	150	92.09 ± 25.98	86.07 ± 32.56	77.18 ± 33.63	48.60 ± 43.37	25.22 ± 41.76	20.06 ± 39.45	20.01 ± 39.44	20.01 ± 39.44	20.01 ± 39.44
30	50	90.71 ± 28.08	93.51 ± 23.41	85.6 ± 30.86	53.80 ± 42.22	24.29 ± 41.28	20.08 ± 39.55	20.00 ± 39.57	20.00 ± 39.57	20.00 ± 39.57
	75	90.27 ± 28.68	94.66 ± 21.01	88.26 ± 27.74	55.26 ± 41.87	24.95 ± 41.35	20.12 ± 39.49	20.01 ± 39.50	20.01 ± 39.50	20.01 ± 39.50
	150	92.28 ± 25.79	93.57 ± 23.04	84.69 ± 32.12	56.42 ± 41.97	25.87 ± 41.92	20.11 ± 39.51	20.01 ± 39.52	20.01 ± 39.51	20.01 ± 39.52
45	50	**98.45 ± 11.10**	**99.60 ± 3.39**	**96.32 ± 16.49**	58.93 ± 42.12	23.51 ± 40.67	20.15 ± 39.74	20.02 ± 39.74	20.02 ± 39.74	20.02 ± 39.74
	75	**98.71 ± 9.79**	**99.54 ± 3.50**	**96.17 ± 16.57**	61.29 ± 41.73	24.17 ± 41.10	20.15 ± 39.67	20.02 ± 39.68	20.02 ± 39.68	20.02 ± 39.68
	150	**98.75 ± 9.48**	**99.15 ± 6.51**	**95.63 ± 18.00**	65.02 ± 40.97	26.58 ± 42.16	20.19 ± 39.62	20.02 ± 39.61	20.02 ± 39.61	20.02 ± 39.61
60	50	92.79 ± 24.74	96.94 ± 15.77	94.69 ± 21.10	60.42 ± 43.90	24.38 ± 41.21	20.18 ± 39.80	20.01 ± 39.80	20.01 ± 39.80	20.01 ± 39.80
	75	92.72 ± 24.80	97.65 ± 13.48	95.17 ± 19.97	64.84 ± 42.76	24.69 ± 41.23	20.22 ± 39.80	20.02 ± 39.79	20.02 ± 39.79	20.02 ± 39.79
	150	93.96 ± 22.77	95.05 ± 20.27	94.35 ± 21.36	68.45 ± 41.81	27.17 ± 42.52	20.24 ± 39.72	20.02 ± 39.70	20.02 ± 39.70	20.02 ± 39.70

Table 6
SD in (%) for channel model training selection in crescent order.

δ_8	8.31
δ_{11}	8.34
δ_9	8.91
δ_2	9.02
δ_1	9.22
δ_7	9.44
δ_5	9.59
δ_3	10.05
δ_{10}	10.11
δ_4	10.49
δ_6	18.06
δ_{12}	12.09

The difference between the expected hit rate in each SNR value and the classification on non-amputee exercise 2 and amputee exercise 1 and 2 is presented in a graphic for each class.

3. Results

In Table 5 is presented the results for the classification for all individuals from the non-amputee database, for all possible configurations on Table 2 trained with the Training Data C from Table 3. The results from Table 5 are used to select the series length N, and are coherent with autocorrelation values presented on Table 4.

To select the channel used to train the general model, the SD for all models is presented in Table 6. The SD's were calculated over the data classified from all 12 models considering all LSTM configurations and times series with length $N = 45$. The Levene test show that the variances are not homogeneous with a p-value equal to 2.2×10^{-16} and w-value equal to 32.18. Considering that model 8 has the lower SD, is the one chosen to classify all testing data.

Sixty GLM procedure were made for the classification hit rate in all 12 channels and 5 classes combinations, using the model trained with the channel 8 and using Training Data C. The hypothesis tested were the difference between means in LSTM and SNR interval for each channel-class combination. The frequency of p-value greater than 0.05 for the different factors of LSTM was 56, showing no relevant difference between the means.

In the cases were $p < 0.05$, two were found for PLI class (channels 7 and 9) and other two for WGN (channels 8 and 11). In Fig. 6 is presented a graphical interaction between LSTM and SNR interval for Channel 7

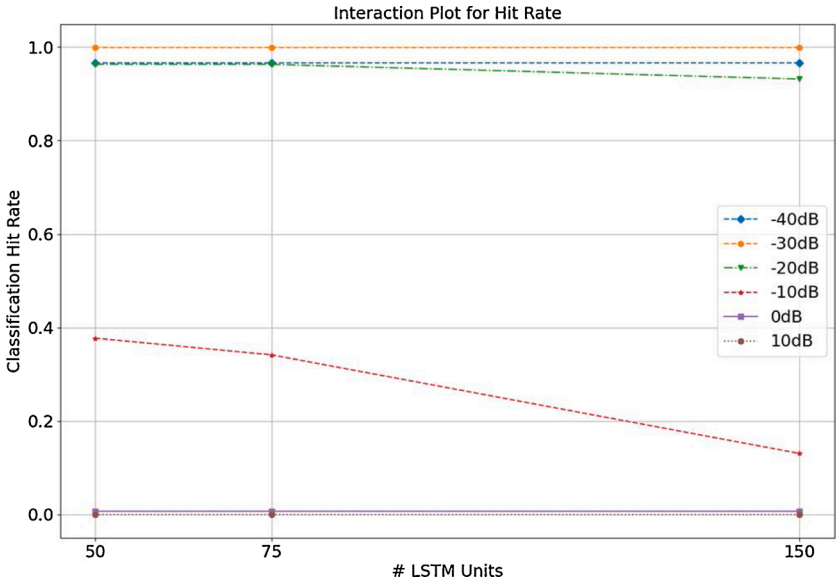

Fig. 6. Result from the GLM analysis for channel 7 and PLI class.

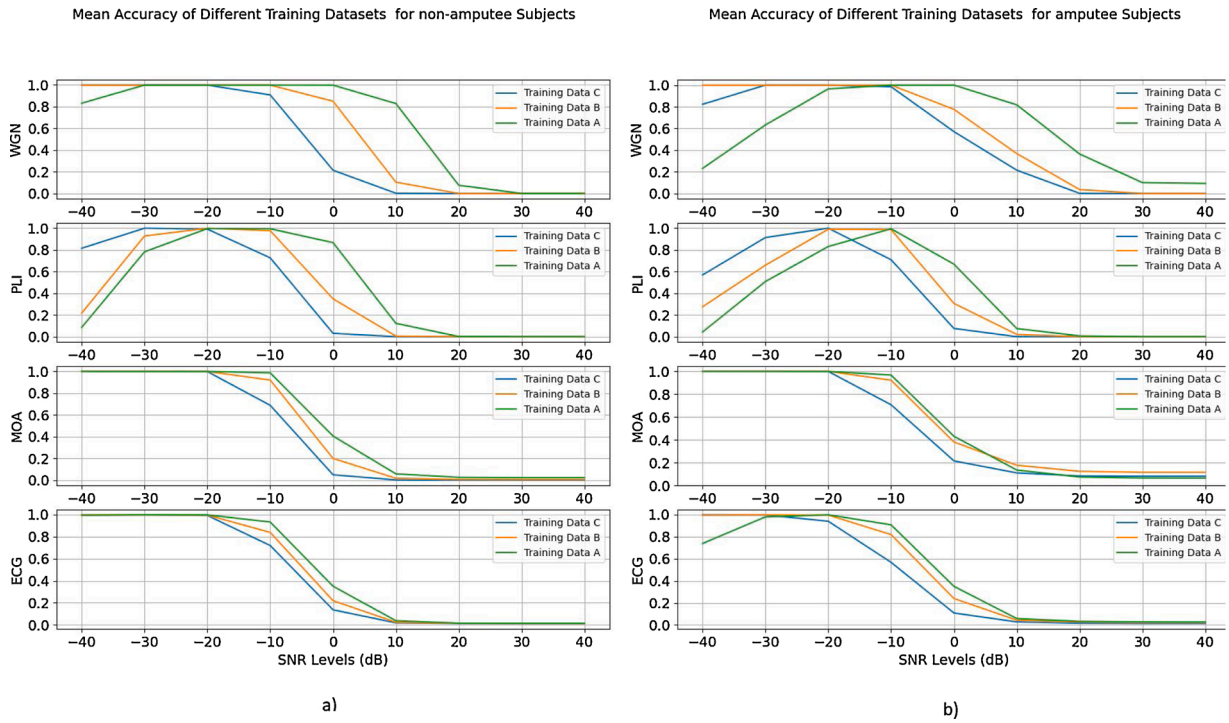

Fig. 7. Fraction of the correctly classified data for all contaminants type and the three SNR data training. (a) for non-amputee subjects and (b) for amputee subjects.

Table 7
Training Data B Confusion Matrix (in %) for Non-Amputee Subjects for a SNR ranging from −40 to 20 dB in 20 dB steps.

SNR			Non-Amputee Subjects				
			Predicted				
			EMG	WGN	PLI	MOA	ECG
−40 dB	True	EMG	**98.49 ± 8.84**	0.05 ± 0.08	0.06 ± 0.41	0.09 ± 0.46	1.31 ± 8.7
		WGN	0.07 ± 0.09	**95.21 ± 19.89**	0.02 ± 0.06	4.69 ± 19.89	0.0 ± 0.01
		PLI	0.00 ± 0.00	0.06 ± 0.09	**41.74 ± 49.0**	58.2 ± 49.01	0.0 ± 0.00
		MOA	0.00 ± 0.00	0.00 ± 0.00	0.03 ± 0.06	**99.91 ± 0.1**	0.06 ± 0.09
		ECG	0.32 ± 0.15	0.00 ± 0.00	0.0 ± 0.00	0.08 ± 0.09	**99.6 ± 0.18**
−20 dB	True	EMG	**98.48 ± 8.84**	0.06 ± 0.09	0.06 ± 0.41	0.09 ± 0.46	1.31 ± 8.7
		WGN	0.06 ± 0.09	**99.87 ± 0.13**	0.06 ± 0.09	0.0 ± 0.00	0.0 ± 0.01
		PLI	0.00 ± 0.00	0.06 ± 0.08	**99.89 ± 0.13**	0.04 ± 0.09	0.0 ± 0.00
		MOA	0.01 ± 0.14	0.0 ± 0.00	0.1 ± 0.09	**99.81 ± 0.19**	0.07 ± 0.11
		ECG	0.66 ± 1.51	0.0 ± 0.00	0.0 ± 0.00	0.11 ± 0.36	**99.23 ± 1.54**
0 dB	True	EMG	**98.5 ± 8.84**	0.03 ± 0.08	0.06 ± 0.41	0.09 ± 0.46	1.31 ± 8.7
		WGN	29.38 ± 44.65	**70.56 ± 44.64**	0.02 ± 0.05	0.04 ± 0.26	0.0 ± 0.01
		PLI	72.76 ± 36.36	0.05 ± 0.15	**25.92 ± 35.3**	0.98 ± 4.37	0.29 ± 1.98
		MOA	87.66 ± 25.62	0.0 ± 0.04	0.02 ± 0.14	**12.24 ± 25.39**	0.07 ± 0.35
		ECG	78.54 ± 23.11	0.0 ± 0.07	0.01 ± 0.09	0.14 ± 0.71	**21.31 ± 22.85**
20 dB	True	EMG	**98.54 ± 8.83**	0.0 ± 0.04	0.06 ± 0.41	0.09 ± 0.46	1.31 ± 8.69
		WGN	99.33 ± 3.14	**0.01 ± 0.08**	0.0 ± 0.03	0.66 ± 3.14	0.0 ± 0.01
		PLI	98.55 ± 8.88	0.01 ± 0.08	**0.05 ± 0.3**	0.11 ± 0.48	1.29 ± 8.66
		MOA	98.49 ± 8.98	0.0 ± 0.04	0.06 ± 0.35	**0.12 ± 0.61**	1.33 ± 8.77
		ECG	98.42 ± 9.29	0.0 ± 0.00	0.05 ± 0.38	0.08 ± 0.41	**1.45 ± 9.15**

and PLI class.

For the test on the SNR-sensitive feature, the results using all training data set from Table 3 for the amputee and non-amputee subjects (exercise 1) is presented in Fig. 7 as fractions of correctly classified data for all SNR range from all subjects and channels. Using the model trained with the Training Data B as example, confusion matrix with the mean value and standard deviation for the fraction of each data type in percentage and with the SNR values ranging from −40 dB to 20 dB in 20 dB steps is presented in for the non-amputee and amputee subjects in Tables 7 and 8.

In Figs. 8 and 9 are presented the difference between the expected SNR hit rate for the standard (non-amputee, exercise 1 results) and the hit rate for the non-amputee exercise 2 and amputee exercise 1. This result is going to show if its possible to predicted the behavior of the model for each SNR level for the different training datasets. Low values indicate a good prediction on the expected behavior, and negatives values above 10 dB indicate a false-positive for contamination.

4. Discussion

Sequence length is a critical parameter for a data-series specialized network and thus was the first parameter analyzed using the results from

Table 8
Training Data B Confusion Matrix (in %) for Amputee Subjects for a SNR ranging from -40 to 20 dB in 20 dB steps.

SNR			Amputee Subjects				
			Predicted				
			EMG	WGN	PLI	MOA	ECG
−40 dB	True	EMG	**86.26 ± 30.27**	0.09 ± 0.08	0.0 ± 0.00	11.47 ± 29.3	2.18 ± 10.03
		WGN	0.02 ± 0.03	**99.93 ± 0.1**	0.03 ± 0.08	0.02 ± 0.07	0.0 ± 0.01
		PLI	0.00 ± 0.00	4.66 ± 20.4	**27.47 ± 44.51**	67.87 ± 46.33	0.0 ± 0.00
		MOA	0.00 ± 0.00	0.0 ± 0.01	0.03 ± 0.06	**99.91 ± 0.08**	0.06 ± 0.08
		ECG	0.26 ± 0.16	0.0 ± 0.00	0.0 ± 0.00	0.44 ± 1.49	**99.31 ± 1.55**
−20 dB	True	EMG	**86.26 ± 30.27**	0.09 ± 0.08	0.0 ± 0.00	11.47 ± 29.3	2.18 ± 10.03
		WGN	0.02 ± 0.03	**99.92 ± 0.1**	0.06 ± 0.11	0.0 ± 0.00	0.0 ± 0.01
		PLI	0.0 ± 0.00	0.1 ± 0.09	**98.82 ± 9.31**	1.08 ± 9.32	0.0 ± 0.00
		MOA	0.0 ± 0.00	0.0 ± 0.00	0.07 ± 0.07	**99.86 ± 0.07**	0.07 ± 0.09
		ECG	0.39 ± 0.41	0.0 ± 0.00	0.0 ± 0.00	0.16 ± 0.62	**99.46 ± 0.77**
0 dB	True	EMG	**86.28 ± 30.28**	0.06 ± 0.08	0.0 ± 0.0	11.47 ± 29.3	2.18 ± 10.03
		WGN	22.39 ± 41.2	**77.6 ± 41.19**	0.0 ± 0.01	0.0 ± 0.0	0.0 ± 0.01
		PLI	52.61 ± 42.67	0.11 ± 0.11	**30.63 ± 35.95**	16.14 ± 33.17	0.52 ± 2.18
		MOA	61.65 ± 42.44	0.0 ± 0.00	0.01 ± 0.02	**38.07 ± 42.07**	0.28 ± 1.11
		ECG	62.08 ± 34.79	0.0 ± 0.00	0.0 ± 0.0	14.0 ± 30.83	**23.92 ± 25.23**
20 dB	True	EMG	**86.34 ± 30.29**	0.0 ± 0.00	0.0 ± 0.00	11.48 ± 29.32	2.18 ± 10.03
		WGN	85.21 ± 27.92	**3.61 ± 18.1**	0.0 ± 0.00	10.59 ± 21.93	0.58 ± 2.98
		PLI	85.94 ± 30.85	0.0 ± 0.00	**0.0 ± 0.00**	11.8 ± 29.58	2.26 ± 10.33
		MOA	85.5 ± 31.21	0.0 ± 0.00	0.0 ± 0.00	**12.34 ± 29.72**	2.16 ± 9.98
		ECG	85.54 ± 31.41	0.0 ± 0.00	0.0 ± 0.00	11.78 ± 29.75	**2.69 ± 12.62**

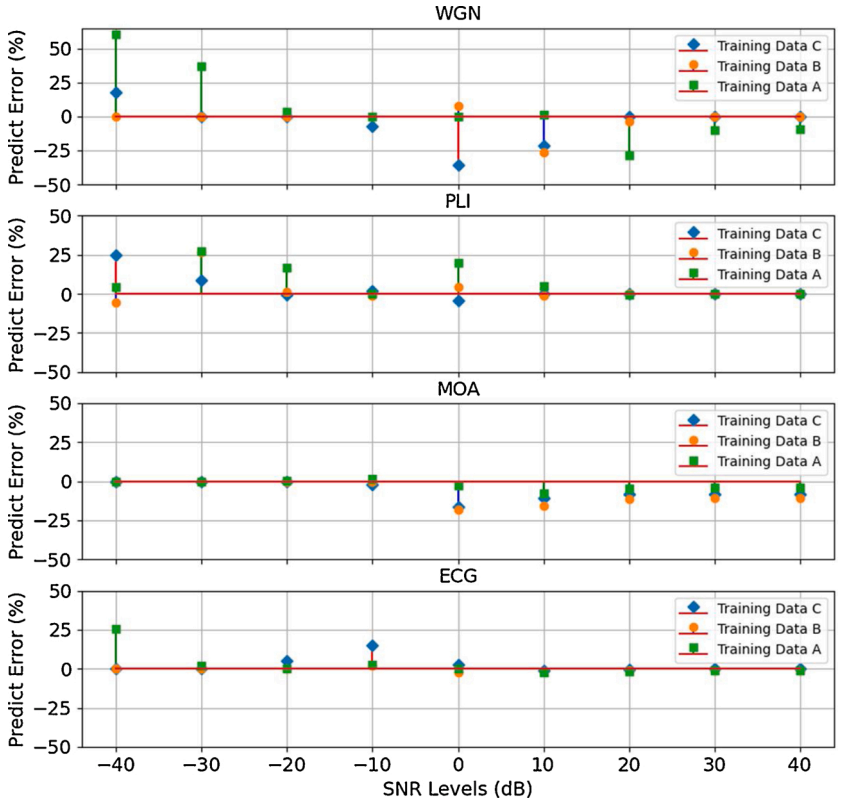

Fig. 8. Difference between the expected value and amputee test subjects, exercise 1.

Table 5. The ideal sequence length is always the shorter one because is computationally more efficient. As could see in Table 5, the lengths different of 45 time-steps presented a standard deviation above 20%, an indication that at least one class is not well characterized by that sequence length size, so the 45 time-steps model is going to be used.

As state in Section II-B, the data from amputee subjects is more problematic than the non-amputee, which is reflect in the results presented in Fig. 7 and Table 8. It is probably that the amputee is contaminated with MOA, as the electrodes cannot be fixed in the ideal locations, but more investigation is needed to analyze the correlation between amputation and more occurrence of MOA in sEMG data.

The WGN impact the sensibility in the amputee data more than in the non-amputee. For the amputee, the signal with 30 dB SNR still got more than 20% identification using Training Data A (the most sensitive to high SNR), whereas in non-amputee the identification is 0%. This could be view as a false contamination detection, because a 30 dB SNR could

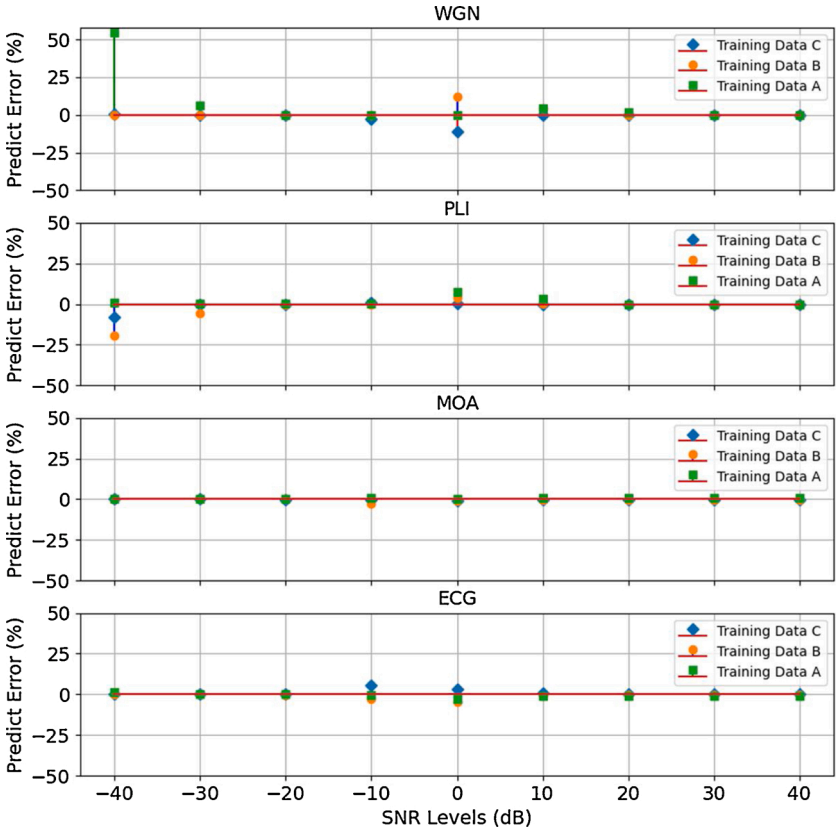

Fig. 9. Difference between the expected value and non-amputee test subjects, exercise 2.

be considered a good quality signal, as indicate in Table 4.

In [35] there are some evidence that an amputation in upper limbs could shift the power spectrum to low frequencies, so the WGN will insert high frequencies components that is detectable even with a high SNR. Also, in our study we found that the power average for all individuals and channels from both database is around 5 time great for frequencies until 500 Hz in the non-amputee database, that could be another reason for the amputee sensibility to WGN.

Using the results from non-amputee, exercise 1 as standard, the Figs. 8 and 9 shows a small error, principally for the non-amputee in all classes, exercise 2 data, except for the −40 dB WGN, Training Data A and all −40 dB PLI for Training Data B and C.

For the amputee, exercise 1, the Training Data A showed too sensitive for WGN, giving more false-positive for contamination above SNR = 10 dB than the others contaminants and it is probably not so adequate to identify WGN in amputee subjects. An error in 0 dB and 10 dB is expected, as the signal is transitioning to non-contaminated.

Compared with two other state of the art works that identify contamination type in sEMG signals [12,16], our outperforms with a 112.5 ms data (5 times 45 time-steps with 2 kHz sample frequency). In [12], three time widow were used: 1 s, 2 s and 5 s with the best results for 2 and 5 s windows and in [16] the time window was 4 s.

Since such contamination can cause an unexpected action from the assistive device, like a prothesis for example, the time window for contamination identification could be critical to avoid dangerous operation.

To classify 5 sequence samples with N = 45 took 5μs and 1 ms for the major voting algorithm (*mode* function from *scipy.stats* Python library) in a Windows® PC with AMD Ryzen® 5 CPU and a NVidia RTX 2060 Super® GPU.

Farrell and Weir [36] states that the optimal processing delay without significantly decreasing prosthesis performance is around 100 ms, and thus, the application based in SOM developed by Ijaz and Choi [16] could be impractical for a multi-channel application, as the processing time is around 25 ms.

Some sEMG PR based system are robust to noise [37] or have a mitigation procedure [13,15]. These techniques could be improved if the detector has a SNR level control, because not all contamination levels will affect the system. In Fig. 7 is clear that WGN and PLI are more affected by the SNR range of training data and MOA and ECG are less sensitive, but as can see in the contamination with −10 dB, the classification is improved from 60% to almost 100% detection.

The PLI contamination is the most unstable outside the training SNR range, been often classified as MOA, as shown in Table 8, but can differentiate MOA and ECG that is known issue in systems build for MOA mitigation in ECG signals [38,39].

5. Conclusions

In this study was presented a novel method for contamination detection in sEMG that has no need of feature extraction and has a SNR sensibility control that provide a small predict error for Training Data B and C in most cases (see Figs. 8 and 9). Also, the use of an upper limb sEMG with amputee and non-amputee subjects was a novelty. The proposed system could be used as a general contaminant detector if turn to a binary classifier with a small amount of false negative, considering the non-contaminated as the positive class.

One weak point from all supervised training models for contaminant identification is that not all possible contaminants are known in advance and in our work we decide to use the most common ones found in the sEMG quality analysis literature [7–16]. One way to get around this limitation is to use a non-supervised method to identify the different contaminations that is presented in the signal, like the one implemented by Ijaz and Choi [16], and then use this data to train a supervised method.

The methodology presented by our study found promising results

that is in accordance with the current literature [12,15,16] and advance the field proposing a faster algorithm by using the raw signal instead of a set of features.

CRediT authorship contribution statement

Juliano Machado: Performed all the signal processing and classification, designed the model architecture, all the methodology, collect the Movement Artifact data and wrote and revised the paper. **Amauri Machado:** Performed the statistical analysis, helped with discussion and conclusions about the statistical results and wrote and revised the paper. **Alexandre Balbinot:** Coordinated the project, providing ideas for the signal processing and classification and wrote and revised the paper.

Declaration of Competing Interest

The authors report no declarations of interest.

Appendix A. Supplementary data

Supplementary material related to this article can be found, in the online version, at doi:https://doi.org/10.1016/j.bspc.2021.102752.

References

[1] J.Y. Hogrel, Clinical applications of surface electromyography in neuromuscular disorders, Neurophysiol. Clin. 35 (2–3) (2005) 59–71, https://doi.org/10.1016/j.neucli.2005.03.001.

[2] J.J. Rasouli, et al., Utility of intraoperative electromyography in placing C7 pedicle screws, J. Neurosurg. Spine (January) (2020) 1–9, https://doi.org/10.3171/2019.11.SPINE191120.

[3] S.M. Engdahl, B.P. Christie, B. Kelly, A. Davis, C.A. Chestek, D.H. Gates, Surveying the interest of individuals with upper limb loss in novel prosthetic control techniques, J. Neuroeng. Rehabil. 12 (December (1)) (2015) 53, https://doi.org/10.1186/s12984-015-0044-2.

[4] A. Gijsberts, M. Atzori, C. Castellini, H. Müller, B. Caputo, Movement error rate for evaluation of machine learning methods for sEMG-based hand movement classification, IEEE Trans. Neural Syst. Rehabil. Eng. 22 (July (4)) (2014) 735–744, https://doi.org/10.1109/TNSRE.2014.2303394.

[5] O. Fukuda, J. Arita, T. Tsuji, An EMG-controlled omnidirectional pointing device, Syst. Comput. Japan 37 (April (4)) (2006) 55–63, https://doi.org/10.1002/scj.20401.

[6] G. Biagetti, P. Crippa, L. Falaschetti, S. Orcioni, C. Turchetti, Human activity monitoring system based on wearable sEMG and accelerometer wireless sensor nodes 08 information and computing sciences 0801 artificial intelligence and image processing 10 technology 1005 communications technologies, Biomed. Eng. Online 17 (November (1)) (2018), https://doi.org/10.1186/s12938-018-0567-4.

[7] A. Arvidsson, A. Grassino, L. Lindstrom, Automatic selection of uncontaminated electromyogram as applied to respiratory muscle fatigue, J. Appl. Physiol. Respir. Environ. Exerc. Physiol. 56 (March (3)) (1984) 568–575, https://doi.org/10.1152/jappl.1984.56.3.568.

[8] C. Sinderby, L. Lindstrom, A.E.E. Grassino, Automatic assessment of electromyogram quality, J. Appl. Physiol. 79 (November (5)) (1995) 1803–1815, https://doi.org/10.1152/jappl.1995.79.5.1803.

[9] C.J. De Luca, L. Donald Gilmore, M. Kuznetsov, S.H.S.H. Roy, Filtering the surface EMG signal: movement artifact and baseline noise contamination, J. Biomech. 43 (May (8)) (2010) 1573–1579, https://doi.org/10.1016/J.JBIOMECH.2010.01.027.

[10] G.D. Fraser, A.D.C. Chan, J.R. Green, D.T. Macisaac, Automated biosignal quality analysis for electromyography using a one-class support vector machine, IEEE Trans. Instrum. Meas. 63 (December (12)) (2014) 2919–2930, https://doi.org/10.1109/TIM.2014.2317296.

[11] G.D. Fraser, A.D.C. Chan, J.R. Green, D.T. Macisaac, Biosignal quality analysis of surface EMG using a correlation coefficient test for normality, MeMeA 2013 - IEEE International Symposium on Medical Measurements and Applications, Proceedings (2013) 196–200, https://doi.org/10.1109/MeMeA.2013.6549735. May.

[12] P. McCool, G.D. Fraser, A.D.C. Chan, L. Petropoulakis, J.J. Soraghan, Identification of contaminant type in surface electromyography (EMG) signals, IEEE Trans. Neural Syst. Rehabil. Eng. 22 (July (4)) (2014) 774–783, https://doi.org/10.1109/TNSRE.2014.2299573.

[13] X. Zhang, H. Huang, A real-time, practical sensor fault-tolerant module for robust EMG pattern recognition, J. Neuroeng. Rehabil. 12 (February (1)) (2015) 18, https://doi.org/10.1186/s12984-015-0011-y.

[14] J. Furukawa, T. Noda, T. Teramae, J. Morimoto, Fault tolerant approach for biosignal-based robot control, Adv. Robot. 29 (April (7)) (2015) 505–514, https://doi.org/10.1080/01691864.2014.996603.

[15] K. de O. A. de Moura, A. Balbinot, Virtual sensor of surface electromyography in a new extensive fault-tolerant classification system, Sensors (Switzerland) 18 (May (5)) (2018) 1388, https://doi.org/10.3390/s18051388.

[16] A. Ijaz, J. Choi, Anomaly detection of electromyographic signals, IEEE Trans. Neural Syst. Rehabil. Eng. 26 (April (4)) (2018) 770–779, https://doi.org/10.1109/TNSRE.2018.2813421.

[17] S. Abbaspour, A. Fallah, M. Lindén, H. Gholamhosseini, A novel approach for removing ECG interferences from surface EMG signals using a combined ANFIS and wavelet, J. Electromyogr. Kinesiol. 26 (February) (2016) 52–59, https://doi.org/10.1016/j.jelekin.2015.11.003.

[18] J. Chen, X. Zhang, Y. Cheng, N. Xi, Surface EMG based continuous estimation of human lower limb joint angles by using deep belief networks, Biomed. Signal Process. Control 40 (February) (2018) 335–342, https://doi.org/10.1016/j.bspc.2017.10.002.

[19] X. Zhang, X. Ren, X. Gao, X. Chen, P. Zhou, Complexity analysis of surface EMG for overcoming ECG interference toward proportional myoelectric control, Entropy 18 (April (4)) (2016), https://doi.org/10.3390/e18040106.

[20] M. Atzori, et al., Electromyography data for non-invasive naturally-controlled robotic hand prostheses, Sci. Data 1 (December) (2014) 140053, https://doi.org/10.1038/sdata.2014.53.

[21] M. Atzori, et al., Effect of clinical parameters on the control of myoelectric robotic prosthetic hands, J. Rehabil. Res. Dev. 53 (3) (2016) 345–358, https://doi.org/10.1682/JRRD.2014.09.0218.

[22] V.H. Cene, A. Balbinot, Enhancing the classification of hand movements through sEMG signal and non-iterative methods, Health Technol. (Berl). 9 (August (4)) (2019) 561–577, https://doi.org/10.1007/s12553-019-00315-6.

[23] M. Besomi, et al., Consensus for experimental design in electromyography (CEDE) project: electrode selection matrix, J. Electromyogr. Kinesiol. 48 (October) (2019) 128–144, https://doi.org/10.1016/j.jelekin.2019.07.008.

[24] A. Goldberger, et al., The MIT-BIH normal sinus rhythm database - PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals, Circ. 101 23 (June (13)) (2000) 215–220, https://doi.org/10.13026/C2NK5R.

[25] A. Graves, A.R. Mohamed, G. Hinton, Speech recognition with deep recurrent neural networks, ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, Oct. (2013) 6645–6649, https://doi.org/10.1109/ICASSP.2013.6638947.

[26] E. López, et al., Wind power forecasting based on echo state networks and long short-term memory, Energies 11 (February (3)) (2018) 526, https://doi.org/10.3390/en11030526.

[27] M. Wielgosz, A. Skoczeń, M. Mertik, Using LSTM recurrent neural networks for monitoring the LHC superconducting magnets, Nucl. Instruments Methods Phys. Res. Sect. A Accel. Spectrometers, Detect. Assoc. Equip. 867 (September) (2017) 40–50, https://doi.org/10.1016/J.NIMA.2017.06.020.

[28] N. Srivastava, G. Hinton, A. Krizhevsky, R. Salakhutdinov, Dropout: a simple way to prevent neural networks from overfitting, J. Mach. Learn. Res. 15 (2014) 1929–1958. Accessed: Feb. 21, 2019. [Online]. Available: https://www.cs.toronto.edu/~hinton/absps/JMLRdropout.pdf.

[29] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (November (8)) (1997) 1735–1780, https://doi.org/10.1162/neco.1997.9.8.1735.

[30] F.A. Gers, J. Schmidhuber, F. Cummins, Learning to forget: continual prediction with LSTM, Neural Comput. 12 (October (10)) (2000) 2451–2471, https://doi.org/10.1162/089976600300015015.

[31] K. Greff, R.K. Srivastava, J. Koutnik, B.R. Steunebrink, J. Schmidhuber, LSTM: a search space odyssey, IEEE Trans. Neural Networks Learn. Syst. 28 (October (10)) (2017) 2222–2232, https://doi.org/10.1109/TNNLS.2016.2582924.

[32] J.T. Gwin, K. Gramann, S. Makeig, D.P. Ferris, Removal of movement artifact from high-density EEG recorded during walking and running, J. Neurophysiol. 103 (June (6)) (2010) 3526–3534, https://doi.org/10.1152/jn.00105.2010.

[33] D.E. Rumelhart, G.E. Hinton, R.J. Williams, Learning representations by back-propagating errors, Nature 323 (October (6088)) (1986) 533–536, https://doi.org/10.1038/323533a0.

[34] D.P. Kingma, J.L. Ba, Adam: a method for stochastic optimization, International Conference on Learning Representations (ICLR) (2015), p. 13.

[35] P.A. O'Neill, E.L. Morin, R.N. Scott, Myoelectric signal characteristics from muscles in residual Upper limbs, IEEE Trans. Rehabil. Eng. 2 (4) (1994) 266–270, https://doi.org/10.1109/86.340871.

[36] T.R. Farrell, R.F. Weir, The optimal controller delay for myoelectric prostheses, IEEE Trans. Neural Syst. Rehabil. Eng. 15 (March (1)) (2007) 111–118, https://doi.org/10.1109/TNSRE.2007.891391.

[37] G.W. Favieiro, A. Balbinot, Paraconsistent random forest: an alternative approach for dealing with uncertain data, IEEE Access 7 (2019) 147914–147927, https://doi.org/10.1109/ACCESS.2019.2946256.

[38] P.S. Hamilton, M.G. Curley, R.M. Aimi, C. Sae-Hau, Comparison of methods for adaptive removal of motion artifact, Comput. Cardiol. (2000) 383–386, https://doi.org/10.1109/cic.2000.898537.

[39] J.D. Costa Junior, J.M. de Seixas, A.M.F.L. Miranda de Sá, A template subtraction method for reducing electrocardiographic artifacts in EMG signals of low intensity, Biomed. Signal Process. Control 47 (2019) 380–386, https://doi.org/10.1016/j.bspc.2018.09.004.