Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 54-4 (2021) 50–55

Recognition and Characteristics EEG Signals for Flight Control of a Drone

Ivan D. Villegas * Julian R. Camargo *
Cesar A. Perdomo Ch. *

* *Universidad Distrital Francisco José de Caldas, LASER research group, Colombia (e-mail: idvillegasi@correo.udistrital.edu.co, jcamargo@udistrital.edu.co, cperdomo@correo.udistrital.edu.co).*

Abstract: This paper shows the results of a proposal to give orders to a drone using electroencephalographic signals, which are filtered between frequencies of 8 and 13 Hz, also some preprocessing in three different types of classifiers are evaluated: random forest, nearest neighbors and convolutional. The best result was the convolutional type network without any preprocessing and with an accuracy higher than 80% and an average time response of 42.21 ms.

Keywords: Brain-computer interface, Drone, random forests, nearest neighbors, convolutional networks.

1. INTRODUCTION

According to the World Health Organization (WHO), more than 15% of the population suffers from some type of disability OMS (2011) and in Colombia, according to the latest DANE (National Administrative Department of Statistics) registry, more than two million people have some characteristic that hinders the development of daily activities. DANE (2010). An increasingly common activity is the use of unmanned aerial vehicles known as drones, which in some countries even require certificates to be used. With the aim of facilitating access and use of this technology to people with motor difficulties, it is proposed to use electroencephalographic signals to give orders to a drone and thus facilitate its handling(Perdomo et al. (2020)).

Electroencephalography is a brain monitoring technique, which receives signals from the cortex of the brain through electrodes placed on the scalp. In order for a person to be able to control an object through brain activity, a communication system is required that monitors brain activity and is capable of translating it to determine the user's intentions. This system is called Brain-Computer Interface (BCI) and can be classified into two main groups depending on the nature of the input signal: endogenous and exogenous. For this paper we use an endogenous BCI system, which depends on the user's ability to control his electrophysiological activity, such as BCI systems based on motor imagery (sensorimotor rhythms) or slow cortical potentials (SCP).HORNERO et al. (2012).

With the current development of EEG technology and the growth of drones, the present paper is oriented to the detection and processing of electroencephalographic (EEG) signals obtained by brain-to-computer interfaces (BCI), with which a series of commands will be generated to send orders to a drone. To achieve this, a structure is proposed as shown in Fig 1, in which the signals taken from the brain are passed to a computer, which is in charge of processing the signals and classifying them to then give a specific instruction to the drone.

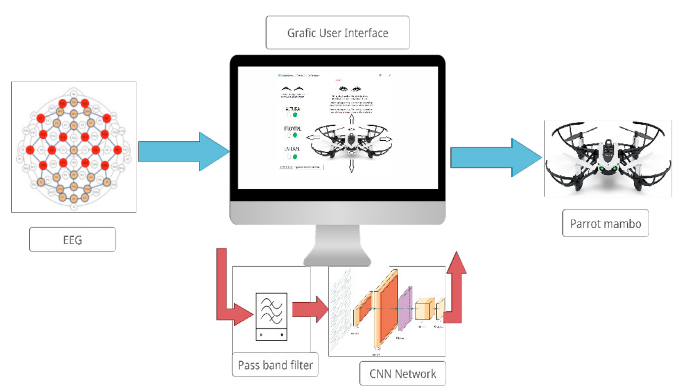

Fig. 1. Overview of the prototype

2. METHODOLOGY

To achieve the objective, it is suggested to use the motor execution paradigm, making use of actions that tend to be discarded because they are sometimes taken by artifacts (electrical potential not originated in the brainLópez et al. (2014).)being blinks and eyebrow raising intentional by the user.For this purpose, the electrodes are distributed according to the international standard 10/20, seeking to cover the greatest amount of information in the motor areas of the cerebral cortex (González Velasco (2012)), taking advantage of the overlapping of the signals due to the fact that the electrodes are located on the scalp in the positions: Fp1, Fp2, C3, C4, Cp5, Cp6, Fc5, Fc6, F7, F8, Fc1, Fc2, T3, T4, Cp1, Cp2 and biauricular references (SDR left ear and BIAS right ear). The electrodes used are dry electrodes of silver chloride silver (Ag-AgCl) mounted on the Ultracortex Mark IV which send the signals to the Cyton and Deisy modules of the OpenBCI company with a sampling frequency of $125Hz$.(OpenBCI (2020)).

10.1016/j.ifacol.2021.10.009

Giving rise to the study of the signals, it is known that EEG signals are composed of different frequency ranges called rhythms, which are associated with different states of the person. For the project, the *alpha* rhythm is used, which ranges from 8 Hz to 13 Hz and is related to a state of consciousness, but not of concentration Sanei and Chambers (2007), Arriola (2016). To isolate this band from the rest, a one-dimensional digital band-pass filter with butterworth topology is implemented in each of the channels.

The recordings for classifier training are taken from people of both sexes, in an age range from 18 to 30 years old, who performed the gestures after listening to a signal provided by the recording software while being supervised by another person in charge of verifying the gesture performed and reporting the gesture to be performed. A GUI designed by the LASER research group was used for data acquisition, allowing the signals to be stored filtered in a specific range(Barreto et al. (2020)).

In some cases it is important to approach the values of the classifier inputs to improve their performance, so we propose to test four different inputs shown in Fig 2; the first one with the filtered signal without preprocessing, an input with the standardized signal after filtering, another scaled between fixed values 0 and 1 after filtering and the last one normalized after filtering (the Python module is used for this *SciKit* Scikit (2020)).

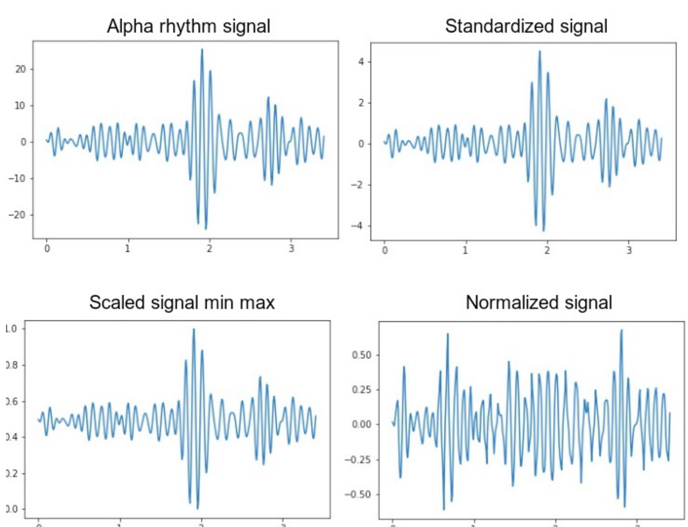

Fig. 2. Signal through the different preprocessing

Motivated by the suggestions of Medina et al. (2018), we try to extract different characteristics in time such as energy or the number of zero crossings, however, when testing these are insufficient to differentiate the gestures. For this reason, the Hjort parameters (also suggested by Rivera Hernández (2016)) are also tested, but no good results are obtained, so direct feature extraction is discarded and it is decided to feed the classifiers with the signals instead of their properties. Following the procedure suggested by Roman (2019), the total data obtained is divided into three groups: training with 70% of the data, validation with 20% of the data and test with 10% of the data. Given the above division, it was decided to implement the following models to catalog the signals in the gestures: the random forest, the k-Nearest Neighborhood and the Convolutional Neural Network, which have 16 channels with 625 data each belonging to the signals captured by the electrodes during 5 seconds and 4 outputs corresponding to the blinking of the left eye with which the drone moves in the negative direction of the selected axis, blinking of the right eye to move in the positive direction of the selected axis, the raising of the eyebrows with which the axis of movement is chosen and the inactivity or execution of another movement with which no action is performed.

The management of the vehicle (parrot mambo) requires the development of an application to connect it with the BCI modules and the classifier. This is developed in python using *PyQT5* for the creation of the interface, *pyOpenBCI* for the connection with the BCI modules and *pyparrot* for the connection and orders of the drone; a thread is used for the connection with the drone and another one with the BCI modules, apart from a timer in charge of executing the classifier and a flag to warn the user when the gesture can be performed.

Once the classifier output is obtained, an algorithm is executed which requests the classifier data and the height, to subsequently evaluate if the gesture belongs to the switch that is going to change the axis of movement or to some command related to the direction. Given the case that the gesture belongs to the one assigned as switch, it checks the state in which the switch is and changes it to the next one, showing which movement is to be enabled to ask again for the next gesture to arrive by the classifier. On the contrary, if the gesture is part of one belonging to the direction, it first asks for the state of the switch, to know in which axis the drone should move and then asks for the gesture to execute the corresponding movement. In case the switch is on the vertical axis, there are two options that will be executed depending on the height at which the drone is located, take off and land if it is close to the ground, or ascend and descend if it is at a certain distance from the ground.

3. RESULTS

The tests carried out for the construction of the CNN classifier have as results the images of Fig. 4, in this one it is obtained that for the first layer the best value is with 6 filters and a *kernel size* of 50, for the second layer the best result obtained is with 16 filters and a *kernel size* of 250, finally for the dense layer the best results are obtained with 250 neurons and the *dropout* of 0.4 and 0.7 with validation percentages higher than 80%.

Fig.3 and Fig.5 show the test results during training, showing the performance according to each metric of the three types of classifiers with the different inputs according to the preprocessing applied.In Fig 3, the results show that the KNN classifier achieves its best performance when the input signal is filtered without any preprocessing, but the results are quite far from each other. In the RF classifier, the best performance is obtained when the input is scaled between fixed values, although its highest value is not equal to that achieved by the KNN, its results have a smaller deviation, being the overall average of the metrics above those of KNN. In Fig. 5 we can see that with

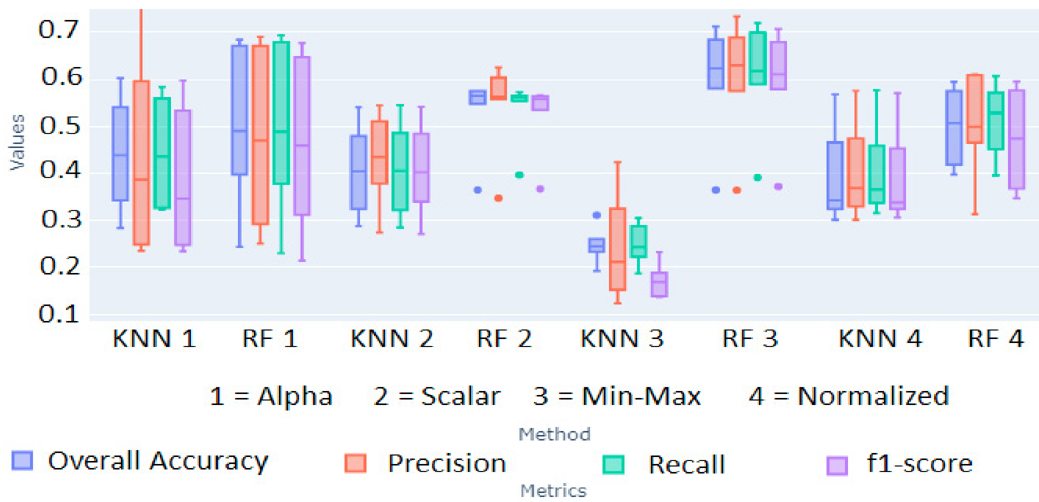

Fig. 3. Test results during KNN and RF classifiers training.

CNN higher results are obtained with the inputs without preprocessing.

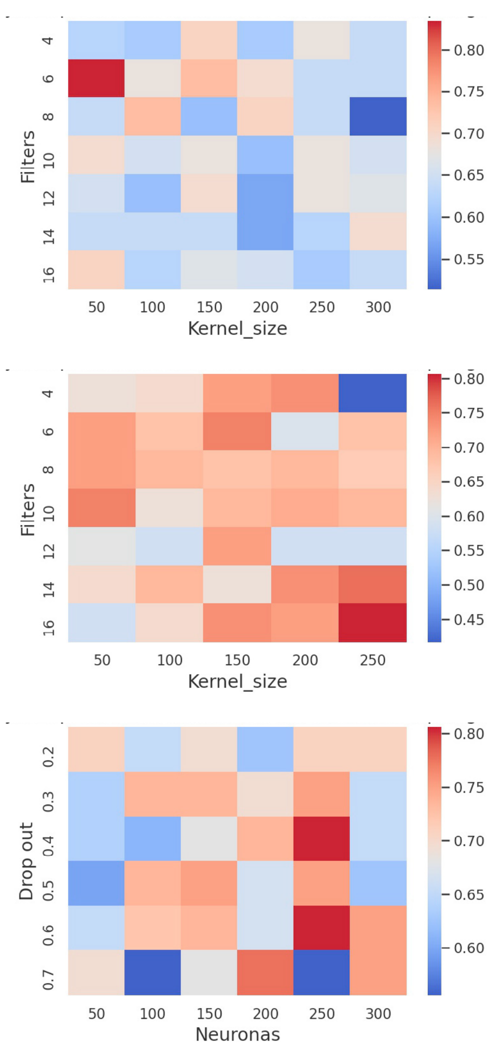

Fig. 4. Heat maps of the tests performed to determine the CNN network values.

Once it was known which of the networks and with which inputs presented the best performance, it was decided to eliminate input channels belonging to electrode pairs and retrain to evaluate how the absence of the electrodes affects the classifier and thus determine which electrode pairs contribute the most information to the network. Table1 shows the results of the experiments organizing the electrode pairs from the most relevant to the least significant for the classifier.

Electrodes removed	Average accuracy [%/100]	Accuracy standard deviation	Average loss	Standard deviation of loss
Fp1-Fp2	0,524	0,045	50,793	91,180
Cp1-Cp2	0,597	0,060	7,109	3,948
C3-C4	0,614	0,031	5,893	1,795
T3-T4	0,614	0,068	5,121	2,527
F7-F8	0,618	0,026	4,143	3,062
Cp5-Cp6	0,642	0,055	4,099	3,478
Fc1-Fc2	0,666	0,071	3,155	1,939
Fc5-Fc6	0,670	0,038	4,321	2,131

Table 1. Average and standard deviation of the tests to determine the importance of each electrode pair in the CNN classifier.

4. DISCUSSION

Regarding the location of the electrodes, it can be said that it fulfills the objective of acquiring enough information for the different actions to be classified; The distribution does not differ much from others suggested, such as that used by López et al. (2014),Botelho et al. (2017) or Batres-Mendoza et al. (2017), so it is not surprising that the electrodes that provide the most information are Fp1 and Fp2, which are located very close to the organs that perform the actions, followed by Cp1 and Cp2, which are the closest to Cz and close to the motor region of the brain, as well as C3 and C4.

The chosen filtering solves several conflicts, being between 7 and 13 Hz there is no need to worry about 50Hz or 60Hz noise, it also avoids electromyographic (EMG) signals, because according to Muthukumaraswamy (2013) the bandwidth of muscle activity is between $\sim$ 20 300Hz; Additionally, this frequency range is within the one used

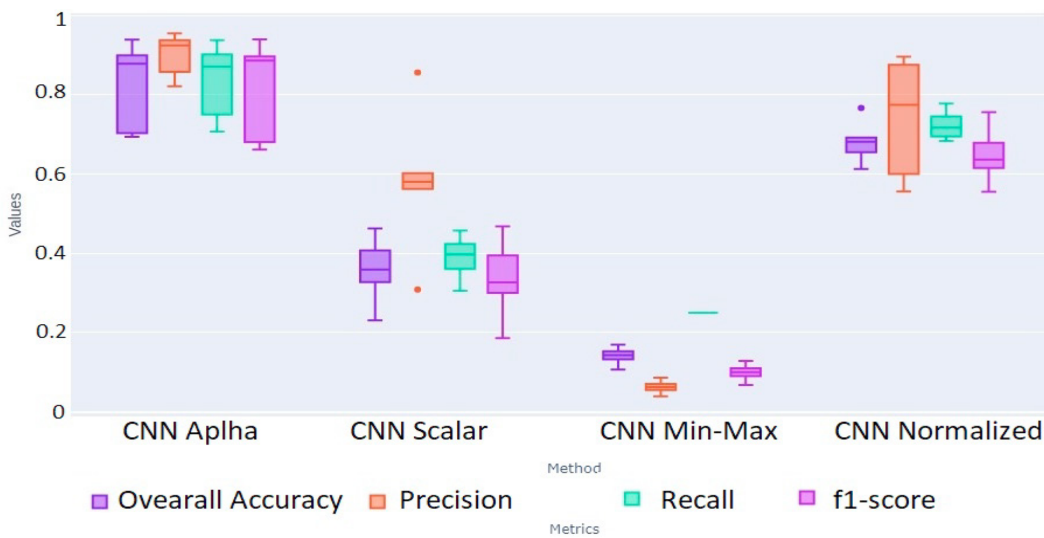

Fig. 5. Test results during CNN classifiers training.

for artifact elimination as shown by López et al. (2014) being quite close to the one shown by b Abd Rani and bt. Mansor (2009) which tells us that EEG signals above 5Hz in areas close to Fp1 and Fp2 show ocular activity very clearly.

When training, validation values higher than 80% are obtained as shown in Fig. 4; if we compare the results obtained in other works such as Batres-Mendoza et al. (2017) where the paradigm is motor imagination which promises values higher than 80% when using a quaternion based signal analysis (QSA) technique, or with Almonacid et al. (2016) which suggest a neuro-fuzzy algorithm with 89.09% of accuracy and an average time response of 42.21ms, the developed model presents itself as a viable option for motor execution with good metrics as shown in Fig. 5. However, the behavior in real time is not as expected.

Fig. 6 shows the confusion matrices obtained by testing the model in two different environments. The upper matrix belongs to the test data, which exhibits excellent behavior in most cases except for eyebrow movement (EB), which is usually confused as inactivity (In); This may be a consequence of the rigidity of the Ultracortex helmet, which in some cases when moving the eyebrows may generate an unwanted movement of some electrodes, generating a noise that is classified as inactivity. It can even happen, that some electrodes obtain additional information, which is not in the flicker signals, but is not sufficiently coincident to be classified in a different class, then it recognizes this signal as inactivity. In the lower matrix you can see the real-time behavior of the network, where you can see that in most cases blinks are still classified correctly; the same is not true for eyebrow raising, now confused with all classes especially with right eye blink (BR) and inactivity which tends to be confused especially with left eye blink (BL). This case may occur because blinking and eyebrow movement have similar signals at the crucial electrodes, so when the eyebrows move, and some electrode has a lower signal amplitude for any reason, for example, as a bad contact, this signal could be classified as a blink and not as an original case.

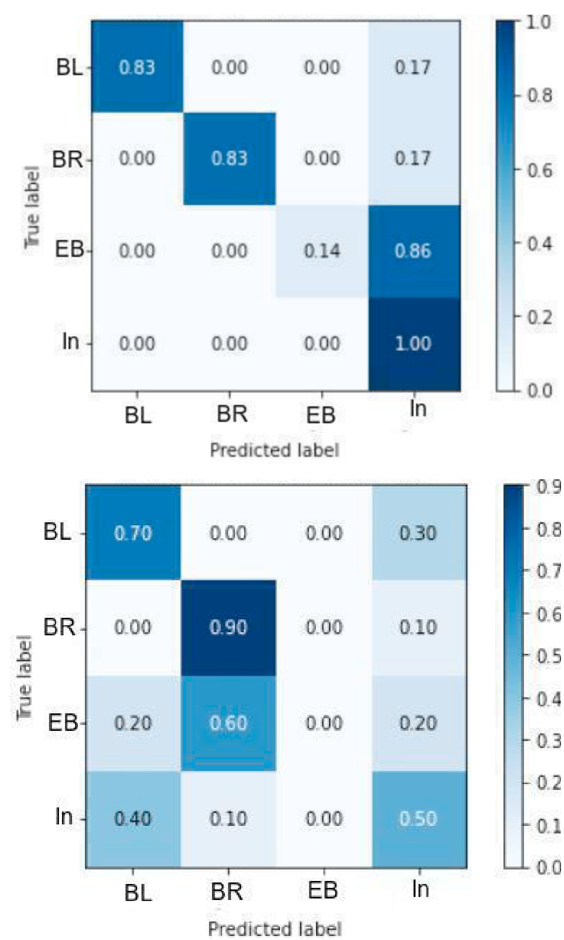

Fig. 6. Confusion matrices from CNN classifier wihtout reprocessing input. The upper image belongs to the evaluation using test data from the database, while the lower one is obtained from real-time tests.

As can be seen in Fig. 6, when implementing the model in real time, there is a drop in its performance, so the metrics in the two matrices are calculated following the suggestions in Shmueli (2019a,b); these are shown in Fig. 7. The graph has as x-axis the different outputs of the

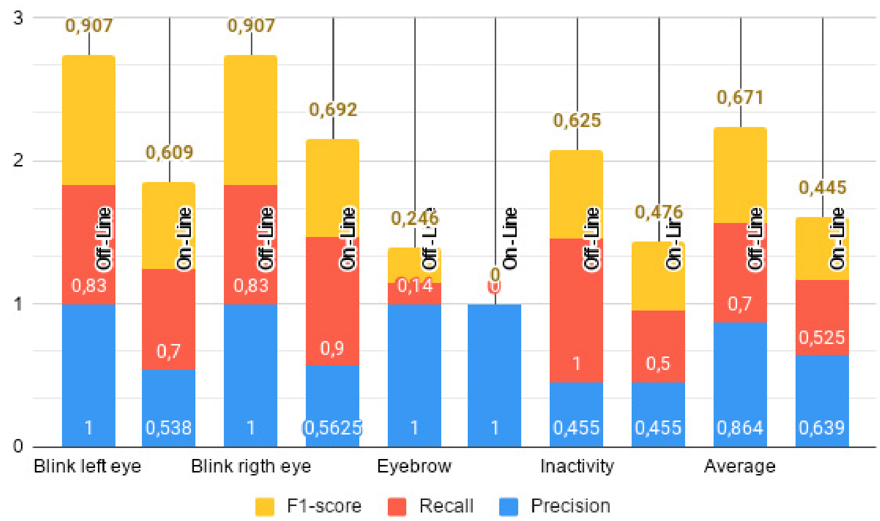

Fig. 7. Comparison of the values of the metrics obtained during the on-line and off-line tests with the two best sets of weights using CNN classifier wihtout reprocessing input.

classifier and an average at the end, in each of these there are two results, one is when testing the network with recordings and the other when testing it in real time. It can be seen that the amount of true positives is quite high in the first two gestures, but when implemented in real time the percentage is reduced to about half, although as the recall shows the amount of false negatives is still similar. The opposite happens with inactivity, the number of false negatives increases decreasing the recall, but the number of true positives remains similar. In summary, true positives decrease from 86.4% to 63.9%, false negatives increase from 30% to 47.5%; on average, the classifier has decreased its performance from 67.1% to 44.5%. This decrease in performance raises an incorporation of other methods alongside the classifier as a possible option to improve performance, such as those proposed by Almonacid et al. (2016) or Batres-Mendoza et al. (2017). It is also possible to consider the RF classifier as an option to consider, since its performance was not tested in real time, but if it behaves as shown in Fig. 3 it may be a better way for real time classification.

The difference between the behavior of the classifier in real time and the test can be influenced by the person wearing the helmet, due to the rigidity of the helmet the data taken varies depending on the person wearing it, regardless of the location of the support points of the ultracortex and the location of the Cyton and Deisy modules, which alter the center of gravity of the device adding pressure on some electrodes.

Despite the difficulties with the classifier, the connection and interaction between the EEG signals and the drone are possible to execute, although with considerable errors in the actions performed, this is because the direction towards which it should move is usually interpreted as inactivity and the change of axis is difficult to perform as it is confused with the direction of movement. So it is worth improving the classifier or implementing one with better performance.

5. CONCLUSION

The distribution of electrodes suggested for the present project, in which 12 electrodes are located around the motor region of the cerebral cortex and 4 near the forehead, fulfills the objective for which it is suggested, since it allows detecting the necessary information to classify with certain precision motor gestures related to the movements of the facial organs. As shown in table 1, since there is no case in which a significantly high precision value is reached, it is evidence that all the electrodes provide information; However, since the electrodes are so close together, there may be a case of redundancy in the information, which opens the possibility that not all the electrodes are necessary to classify the gestures that are worked on in this document, this is said based on the fact that there are certain pairs of electrodes that provide more information than others, such as those located in Fp1, Fp2, Cp1, Cp2, C3, C4, T3 and T4.

Although no tests were performed filtering the signal in a frequency range other than that of the *alpha* waves, it is considered necessary to filter the signal for the classifier to perform its work properly. As the results of the CNN classifier show, the filtered signals are sufficient to adequately perform the classification, but it is not considered to be something that can be ignored, because in the other classifiers that were tested the preprocessing to leave the signals in a similar range of values is necessary to improve the performance of the network, without adequate filtering the network will be unable to differentiate the electrodes where there is brain activity from those where there is only noise, because all the channels have similar values.

Performing a process where the signal data have close values (especially between the fixed values zero and one) shows improvements in the RF and KNN classifiers, in which improvements can be seen with respect to the signal that has only been filtered. On the contrary, where the

CNN classifier worsens the performance of the network if the signals have similar values.

Among the paradigms evaluated, the best performing ones are the RF (with the data filtered at the rate *alpha* and subsequently scaled between values of zero and one) where an accuracy close to 85% is achieved and the CNN network (with the data filtered at the rate *alpha*) where an accuracy of 86.66% is achieved, the difference between the models lies mainly in its response to unknown data, where the CNN network has a better performance achieving 78.25% accuracy with totally unknown data.

To implement the classifier in real time, it is necessary to indicate to the user in which time interval he can perform the gesture, this because several gestures can be performed in the same time interval that confuse the network, which reduces its performance and can also prevent the user from getting confused due to the delay between his commands and what is shown in the GUI. However, it is important to mention that the classifier presents a drop in its performance compared to what was shown in the training and testing of the same in real time, placing the RF classifier as an option to take into account since during the training it showed a good performance and it would be worthwhile to evaluate its behavior in real time.

The development of the GUI facilitates to verify that the algorithm and the classifier work in an acceptable way, since the screen allows to see the actions that should be performed, which in most cases are the desired ones, in cases where not, they are taken as inactivity or as a change in the axis of the movement, However, this does not always occur due to the decrease in the accuracy of the classifier when a left eye blink and eyebrow raise are performed, which are sometimes mistaken for a right eye blink.

REFERENCES

Almonacid, M., Ibarrola, J., and Cano-Izquierdo, J. (2016). Voting strategy to enhance multimodel eeg-based classifier systems for motor imagery bci. *IEEE Systems Journal*, 10(3), 1082–1088. doi:10.1109/JSYST.2014.2360433.

Arriola, J.M. (2016). *Representación matemática de ondas cerebrales*. Master's thesis, UNIVERSIDAD NACIONAL DEL SUR.

b Abd Rani, M.S. and bt. Mansor, W. (2009). Detection of eye blinks from eeg signals for home lighting system activation. In *2009 6th International Symposium on Mechatronics and its Applications*, 1–4. doi:10.1109/ISMA.2009.5164828.

Barreto, F., Daza, N., Sanchez, C., and C.A., P. (2020). Proyecto de investigación open bci y myo armband. URL `https:\/\/github.com\/fabianbarreto02\/Open\_BCI\_and\_MYO\_UD`.

Batres-Mendoza, P., Ibarra-Manzano, M.A., Guerra Hernandez, E.I., Almanza Ojeda, D.L., Montoro Sanjose, C.R., Romero-Troncoso, R.J., and Rostro-Gonzalez, H. (2017). Improving eeg-based motor imagery classification for real-time applications using the qsa method @ONLINE. *Computational Intelligence and Neuroscience, https://www.hindawi.com/journals/cin/2017/9817305/.*

Botelho, T.R., Soprani, D., Rodrigues, C., Ferreira, A., and Frizeraa, A. (2017). Nuevo enfoque para la clasificación de señales eeg usando la varianza de la diferencia entre las clases de un clasificador bayesiano. *Revista Iberoamericana de Automática e Informática Industrial*.

DANE (2010). Población con registro para la localización y caracterización de las personas con discapacidad – informe de investigación (departamento administrativo nacional de estadística) @ONLINE.

González Velasco, A. (2012). El movimiento humano. Technical report, Universidad de Salamanca.

HORNERO, R., CORRALEJO, R., and ÁLVAREZ, D. (2012). Brain-computer interface (bci) aplicado al entrenamiento cognitivo y control domótico para prevenir los efectos del envejecimiento. *Lychnos*, 08.

López, B., Úbeda, A., Planelles, D., Perez, C., Iáñez, E., and Azorín, J. (2014). Aplicación de ica para la eliminación de artefactos oculares y mandibulares en señales eeg durante la realización de tareas mentales. *Actas de las XXXV Jornadas de Automática*. URL `http://www.ja2014.upv.es/wp-content/uploads/papers/paper\_15.pdf`.

Medina, B., Sierra, J.E., and Barrios Ulloa, A. (2018). Técnicas de extracción de características de señales eeg en la imaginación de movimiento para sistemas bci. *Espacios*, 39.

Muthukumaraswamy, S.D. (2013). High-frequency brain activity and muscle artifacts in meg/eeg: a review and recommendations. *Frontiers in human neuroscience*, 7. doi:10.3389/fnhum.2013.00138.

OMS (2011). Resumen informe mundial sobre la discapacidad (organización mundial de la salud) @ONLINE.

OpenBCI (2020). Openbci: Documentation, @ONLINE. https://docs.openbci.com/docs/Welcome.html#docs-website-update.

Perdomo, C., Barreto Sanchez, F., and Camargo, J. (2020). Artificial intelligence implemented in gesture recognitions of the upper limb. *International Journal of Mechanical and Production Engineering Research and Development*, 10, 10617–10626.

Rivera Hernández, F.O. (2016). *Sistema clasificador para ondas cerebrales de acciones simples para la interacción con robots*. Master's thesis, Universidad Veracruzana.

Roman, V. (2019). Machine learning: Cómo desarrollar un modelo desde cero, @ONLINE. *Ciencia y Datos, https://medium.com/datos-y-ciencia/machine-learning-c%C3%B3mo-desarrollar-un-modelo-desde-cero-cc17654f0d48.*

Sanei, S. and Chambers, J.A. (2007). *EEG Signal Processing*. John Wiley and Sons Ltd, primera edición edition.

Scikit (2020). *Preprocessing data.* scikit learn. https://scikit-learn.org/stable/modules/preprocessing.html.

Shmueli, B. (2019a). Multi-class metrics made simple, part i: Precision and recall, @ONLINE. *Towards Data Science, https://towardsdatascience.com/multi-class-metrics-made-simple-part-i-precision-and-recall-9250280bddc2.*

Shmueli, B. (2019b). Multi-class metrics made simple, part ii: the f1-score,@ONLINE. *Towards Data Science, https://towardsdatascience.com/multi-class-metrics-made-simple-part-ii-the-f1-score-ebe8b2c2ca1.*