ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE SISTEMAS

UNIDAD DE TITULACIÓN

DESARROLLO DE UNA INTERFAZ CEREBRO-COMPUTADOR PARA EL CONTROL DE DISPOSITIVOS IOT MEDIANTE EL USO DE SEÑALES EEG DE PARPADEOS PARA PERSONAS CON DISCAPACIDAD MOTRIZ

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN SISTEMAS INFORMÁTICOS Y DE COMPUTACIÓN

KELVIN ALEXIS ORTIZ CHICAIZA

kelvin.ortiz@epn.edu.ec

Director: MARCO E. BENALCÁZAR

marco.benalcazar@epn.edu.ec

Quito, agosto 2021

DIRECTOR'S APPROVAL

As director of the degree work DEVELOPMENT OF A BRAIN-COMPUTER INTERFACE FOR THE CONTROL OF IOT DEVICES THROUGH THE USE OF EYE BLINKS EEG SIGNALS FOR PEOPLE WITH MOTOR DISABILITIES developed by Kelvin Alexis Ortiz Chicaiza, a student of the Faculty of Systems, having supervised the realization of this work and made the corresponding corrections, I consider approved the final draft of the written document so that it can continue with the procedures corresponding to the support of the Oral Defense.

Firmado electrónicamente por:
MARCO ENRIQUE
BENALCAZAR
PALACIOS

MARCO E. BENALCÁZAR

DIRECTOR

AUTHORITY STATEMENT

I, Kelvin Alexis Ortiz Chicaiza, declare under oath that the work described here is my own; that has not previously been submitted for any degree or professional qualification; and, that I have consulted the bibliographic references included in this document.

The National Polytechnic School can make use of the rights corresponding to this work, as established by the Intellectual Property Law, by its Regulations and by current institutional regulations.

Kelvin Alexis Ortiz Chicaiza

DEDICATION

I dedicate this work to my mother who with her love and dedication supported me and motivated me to keep going through difficult times. She taught me that the only way to fulfill my dreams is hard and consistent work. She guided me in the study because she knew that knowledge is the most powerful tool there is. She works tirelessly to maintain our home. She was present in the sad and happy moments of my life teaching me values that I will always bear in mind. She watched me grow up, become a professional and I hope I have fulfilled her dream.

GRATITUDE

I thank my mother for all the unconditional support that she has given me and thanks to her I was able to get to the point where I am.

To my brothers Javier and Jordan, who in them I saw a north and thanks to their constant support and words of encouragement I was able to go further every day.

To Paola, that she knew how to accompany me in my university career, supporting me, advising me, helping me to correct my defects, making me a better man and sharing moments that I will never forget.

To my friends who supported me and gave me many happy moments over the years, making college the best time of my life.

To all my classmates with whom I had the pleasure of sharing a class, a party or an outing.

To my tutor Marco E. Benalcazar for the help, time and knowledge of him, in addition, he knew how to guide me in an excellent way in the development of this project.

To Sang Guun Yoo, for hosting me in the research laboratory that helped me pose new challenges and guide this project.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ANNEXES

ABSTRACT

Motor disability is the loss of the ability to move a limb of the body. Motor disabilities make difficult the interaction between a disabled person and her/his environment. Recent research has focused on developing innovative technologies that could be used by disabled people to improve their life quality. In this paper, a brain-computer interface for controlling IoT devices is proposed. This system is based on the use of the Muse-Headband sensor which captures EEG signals when a person blinks. This sensor is placed on the forehead of the user of the system. The EEGs are preprocessed and then classified into short and long blinks by computing their signal envelopes and using the k-NN algorithm. The classified blinks are then used to form control commands that are sent to a mosquitto server hosted in the cloud. This server is responsible for sending the control action to the connected IoT devices. The accuracy of the classifier designed in this work is 99.53%. Usability tests show that the probability of a user sending a wrong command, with the proposed system, is 4.83%. However, this probability decreases when the time of use of the system proposed increases.

Keywords: EEG, BCI, k-NN, IoT.

ABSTRACT

Motor disability is the loss of the ability to move a limb of the body. Motor disabilities make difficult the interaction between a disabled person and her/his environment. Recent research has focused on developing innovative technologies that could be used by disabled people to improve their life quality. In this paper, a brain-computer interface for controlling IoT devices is proposed. This system is based on the use of the Muse-Headband sensor which captures EEG signals when a person blinks. This sensor is placed on the forehead of the user of the system. The EEGs are preprocessed and then classified into short and long blinks by computing their signal envelopes and using the k-NN algorithm. The classified blinks are then used to form control commands that are sent to a mosquitto server hosted in the cloud. This server is responsible for sending the control action to the connected IoT devices. The accuracy of the classifier designed in this work is 99.53%. Usability tests show that the probability of a user sending a wrong command, with the proposed system, is 4.83%. However, this probability decreases when the time of use of the system proposed increases.

Keywords: EEG, BCI, k-NN, IoT.

1. INTRODUCTION

People with motor disabilities such as muscular dystrophy could find very complicated to perform simple tasks like, for example, turning on a television, thus making hard their interaction with their environment. For many people with motor disabilities, it is difficult to count, all the time, on a person who can provide support to perform their daily activities. For this reason, different alternatives have been proposed over time to provide this support, including the use of brain-computer interfaces (BCI) [1]. The function of a BCI is based on the capture of EEG signals generated by the activity of the Central Nervous System (CNS) [2]. Several BCIs to improve the life of people with motor disabilities have been proposed including intermediaries to control electronic IoT devices [3]–[7], remote controls for computer cursors [8]–[13], spellers through which users can communicate by forming words [14]–[19], systems for the manipulation of objects in 2D and 3D environments [20]–[22], and control systems for prostheses [23], [24], rehabilitation robots [25]–[28] and robotic arms [29].

The existing BCIs use very complex classifiers [5], [7], [10]–[19], [21]–[24], [26], [27], [30]. For this reason, in this work a classifier based on the use of the k-NN algorithm is developed. We chose this algorithm because of its simplicity and low computational cost. In the development of BCIs there are different methods to obtain EEG signals. These methods can demand very long and complex training sessions, suffer from poor signal decoding and their performance vary significantly between different people [31]. To avoid these problems, in this work we use EEG signals measured while a user blinks. These EGG signals are easy to measure, do not require sensors that demand long and complex training for their use [32], have low inter and intra-user variation of their statistical properties [33], [34], and are simple to decode using the proper filters [35]. Additionally, most of the existing BCIs are limited to run on desktop computers [1]–[27], which limits their portability. To circumvent this problem, the system proposed in this work runs on a tablet or cellphone.

The BCI proposed in this work is an alternative to the help that a subject could provide to a person with motor disabilities. The BCI proposed consists of a mobile application in which the IoT devices and the available control commands are displayed. To control the system, the user must send combinations of long and short eye blinks that generate EEG signals, this model was replicated from [36]. These signals are the input of a preprocessing algorithm followed by a k-NN classifier. To control any device that has been registered in the application, a connection between the mobile application and an mosquitto server was implemented. Finally, usability tests were carried out with 5 people.

The remainder of this paper is organized as follows. In section 2, the materials and methods used for this work are presented. In section 3, the tests to evaluate the system proposed together with their results are presented. Finally, in section 4, the conclusions and future work are presented.

1.1. General objective

Develop a brain-computer interface for the control of IoT devices through the use of eye blinks EEG signals for people with motor disabilities.

1.2. Specific objectives

- Develop a graphical user interface in which the IoT devices and the available control commands are displayed.
- Integrate the brain-computer interface to an intelligent classifier of EEG signals and establish an average time below 3 seconds when obtaining and classifying signals that are produced when the user blinks to send control commands.
- Implement a connection between the brain-computer interface and an MQTT server in charge of receiving and sending control commands with response times less than 1 second.
- Carry out 5 usability tests of the system with people without physical limitations and evaluate the results obtained.

2. METHODOLOGY

This section describes the materials and methods used to develop the system. After this section, the results obtained with the classifier performance and system usability tests are described.

2.1. System architecture

The first block of the proposed BCI is the Muse-Headband, which is a sensor that measures EEG signals on the forehead of a user. This headband sends the EEG signals to a mobile device via Bluetooth. The second block is a mobile device which has installed an application developed in this work. This application translates the EEG signals into commands to control IoT devices. The third block is an mosquitto communication server that receives the commands generated by the mobile application and sends these commands to the subscribed IoT devices. The fourth block is an IoT device, which in this work is a raspberry pi 3 single board computer (SBC). This device, subscribed to the mosquitto server, sends infrared signals to control a TV. The diagram of the BCI proposed is shown in Fig. 1 and the architecture was replicated from [37].

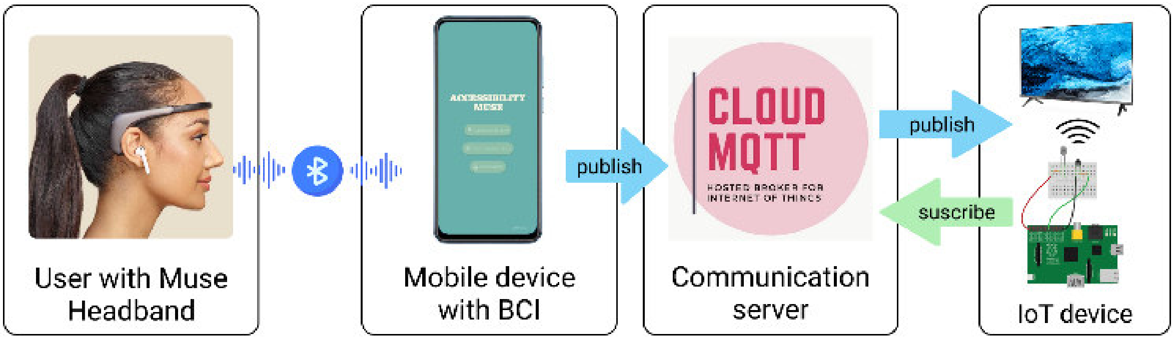


Fig. 1 Diagram of the proposed BCI to control IoT devices.

2.2. Materials

2.2.1. Muse-Headband

A low cost OpenSource sensor for recording EEG signals called Muse-Headband [38] was used for this work. This sensor has a battery that provides electrical autonomy of up to 6 hours of continuous transmission. Because of its small size, the Muse-Headband is a portable device. This sensor has four reception channels at a sampling rate of 255Hz with a resolution of 12 bits.

2.2.2. Android SDK

Android is a software created by Google that allows developers to use its development kit (SDK) with no payment [39]. The android SDK is part of the android development environment (Android Studio). The Android Studio development environment was used to create the mobile application developed in this work.

2.3. Classifier design

In this work, the nonparametric k-NN classifier was used. This algorithm is considered simple, and highly efficient and effective to solve classification problems [40]. k-NN classifies a new example using the labels of the k nearest neighbors, from a training set to the example being classified.

The operation of the k-NN classifier designed in this work follows the steps described in the flow chart shown in Fig. 2. This classifier labels EEG signals with one of 3 classes: Short Blink (SB), Long Blink (LB) and No Blink (NB). The classifier designed uses a dataset composed of 15 samples per class. When an EEG, corresponding to a user blink, needs to be classified, the distance between this signal and the samples in the data set is computed. This distance is computed using the Dynamic Time Warp (DTW) algorithm. DTW first performs an optimal alignment between two input signals represented as time series (time-dependent) [41] and then returns the sum of the absolute value of the pointwise difference between these signals.

The DTW distances between a new EEG and the EEGs in the dataset represent a measure of their similarity. For each EEG in the dataset, its label and the DTW distance between this signal and the new EEG were stored in a list. Then, this list was sorted based on the DTW distances in an ascending order. Next, the mode of the labels of the first k examples of this sorted list was computed. The value of the mode was then divided by k, obtaining thus an estimate of the conditional probability of the new EEG belonging to each of the 3 classes considered in this work. The new EEG was then labeled with the class with the highest probability, provided that this probability value is equal to or greater than a certain threshold U. The values used for this work are k = 5 and U = 0.70. These values were found empirically.

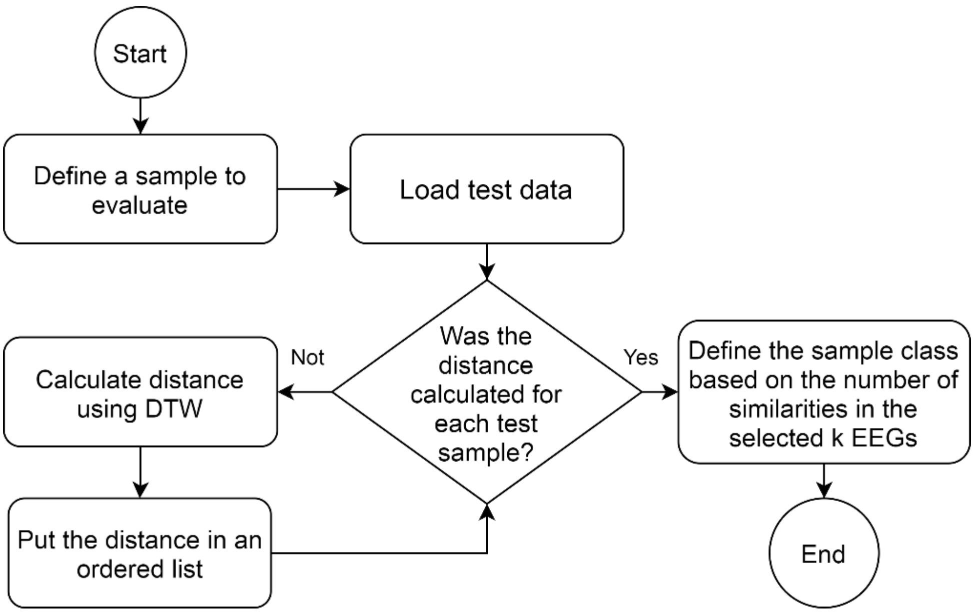


Fig. 2 Flowchart of the classification model designed in this work.

2.4. Data recording

Electroencephalography relies on sensors that detect the electrical activities of the brain that are recorded on the scalp [42]. This work uses the non-invasive technique for the registration of EEG signals. This technique is based on placing electrodes on the scalp of a user to capture EEG signals. The advantages of this technique are the following ones: It does not physically affect the human body, its installation and use is simple and the cost of the equipment is low [6] relative to other techniques.

A dataset of 10 people was recorded and uploaded to an online repository in [43]. For data recording, each person was asked to make a blink for 2 seconds. For each person, a total of 50 EEGs for each of the 3 classes considered in this work were recorded. To have a better description of the data acquired for this work, in addition to the EEGs, we also recorded the age, weight, height and whether the subject felt tired or rested during the data acquisition.

The data of whether a person was tired or rested while data recording is important to register because, if a person is tired, the recorded EEG signal may have a higher frequency [34] than the EEGs of a rested person. The fatigue could affect the frequency content of the signal, thus causing that a blink is not detected by the classifier designed. Age is a determining factor when capturing EEG signals from a person, since these signals have higher average amplitude in young subjects [33]. This amplitude variation affects the detection of blinks by the proposed system, since to detect a blink, the input EEG signal must be equal to or greater than a minimum amplitude value defined by the user. Finally, the Muse-Headband was placed on the forehead of the users, ensuring that the electrodes are at points TP9, AF8, AF8 and TP10 (Fig. 3), following the 10-20 system, a recognized method for placing the electrodes on the scalp [44].

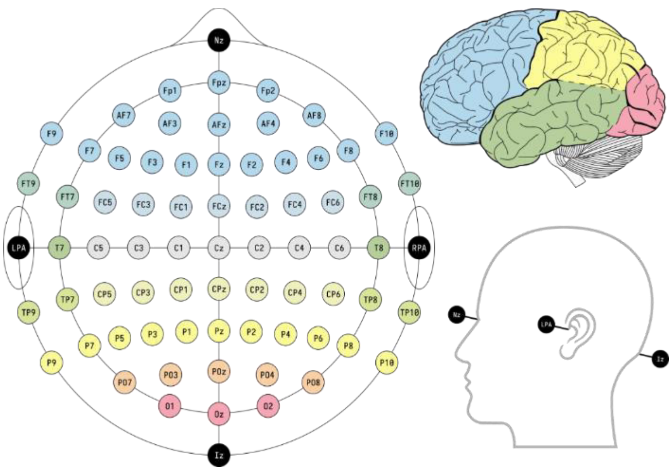


Fig. 3 Location of EEG electrodes according to the 10-20 system

2.5. Signal processing

Processing an EEG signal contributes to improve the classification accuracy. In this work, the EEGs were filtered using a 5th order Butterworth filter, with a cutoff frequency of 2 Hz and a sampling frequency of 256 Hz. This filter was used to eliminate the unwanted parts of the signals in the frequency domain. An example of the result of the filter used in this work is presented in Fig. 4 and Fig. 5.

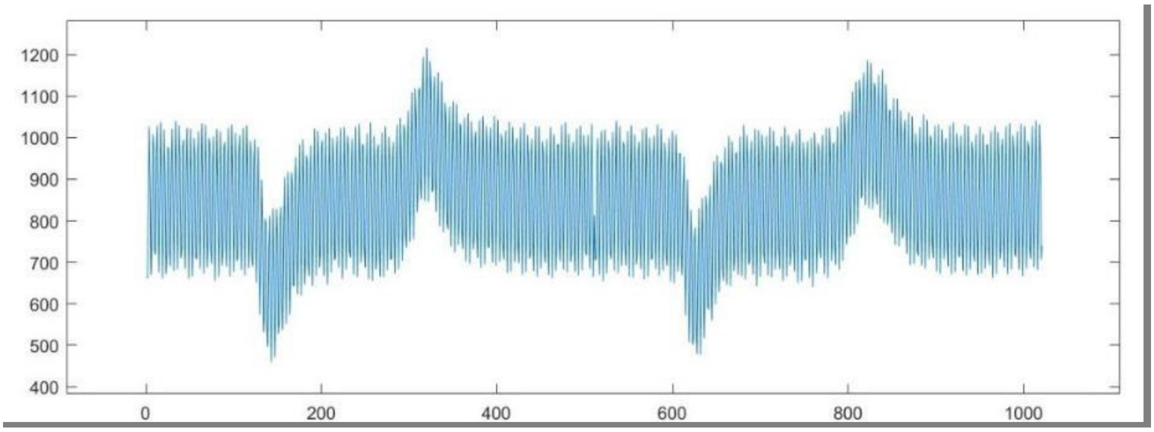


Fig. 4 Unfiltered EEG signal of a long blink.

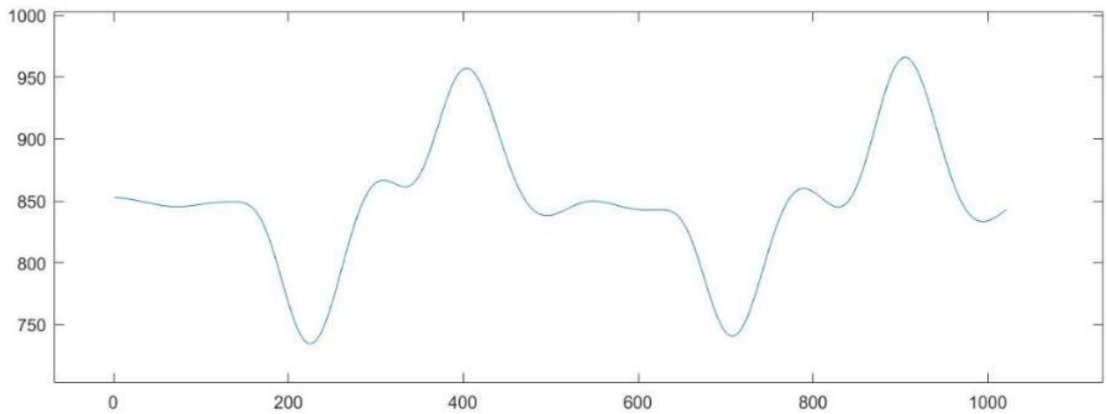


Fig. 5 Filtered EEG signal of a long blink.

The EEG signals of blinks are seasonal since their frequency does not vary much as a function of time [35]. This property reduces the complexity of the task of classifying EEGs of short blinks from long blinks once the EEGs have been filtered. To visualize the spatial distribution of the cloud of points of each class considered in this work, the t-SNE algorithm was used. This algorithm is used for a non-linear dimensionality reduction [45]. Thus, using t-SNE we can visualize each of the processed EEG signals in 2 dimensions. In Fig. 6, the cloud of points of the processed EEGs of each type of blink is presented. We can observe that the cloud of processed EEGs for each class are well separated from each other, simplifying thus the work of the classifier.

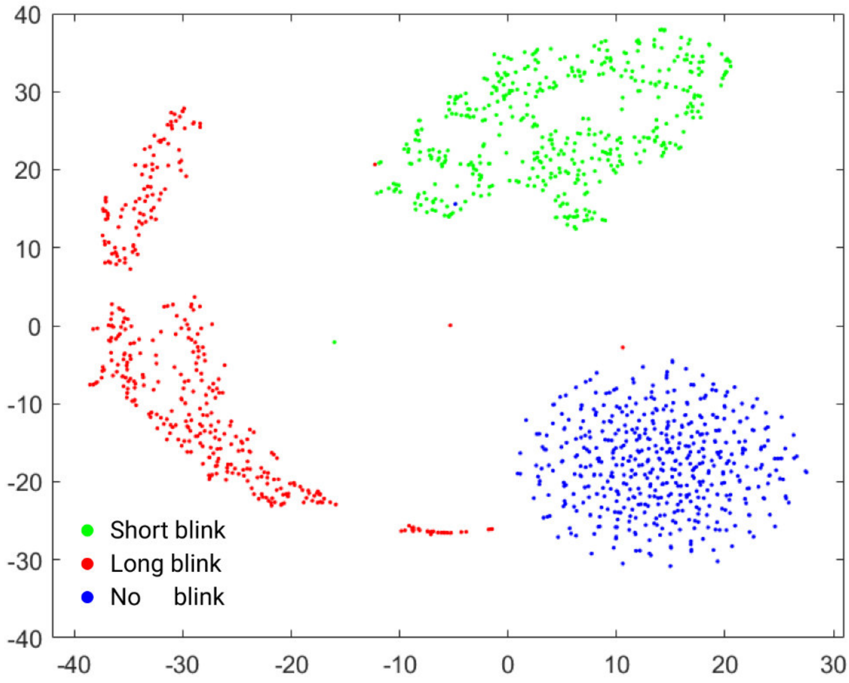


Fig. 6 Cloud of EEGs for each class used in this work. The t-SNE algorithm was used for dimensionality reduction.

2.6. Blink detection

To reduce the appearance of false positives at the detection of blinks, it was necessary to design an algorithm for the detection of blinks (Fig. 7). If a window observation of an EEG passes the detection of a blink, then this signal is sent to the classifier to identify whether it corresponds to a short or long blink. Otherwise, the window observation is labeled as no blink. The following steps are applied for blink detection.

- A circular buffer with a size of 510 points is created.
- A 15-point window is created. This window contains 15 points back from point P.
- The mean amplitude of the signal is calculated, which corresponds to the average of the values inside the 15-point window.
- A comparison between the mean amplitude of the 15-point window and a threshold, AU which was defined by the user, is performed.
- If the mean amplitude inside the window is less than the threshold, then the arrival of 390 new points is expected from point P.

- After the 390 points have arrived, the entire buffer is sent to the classifier.

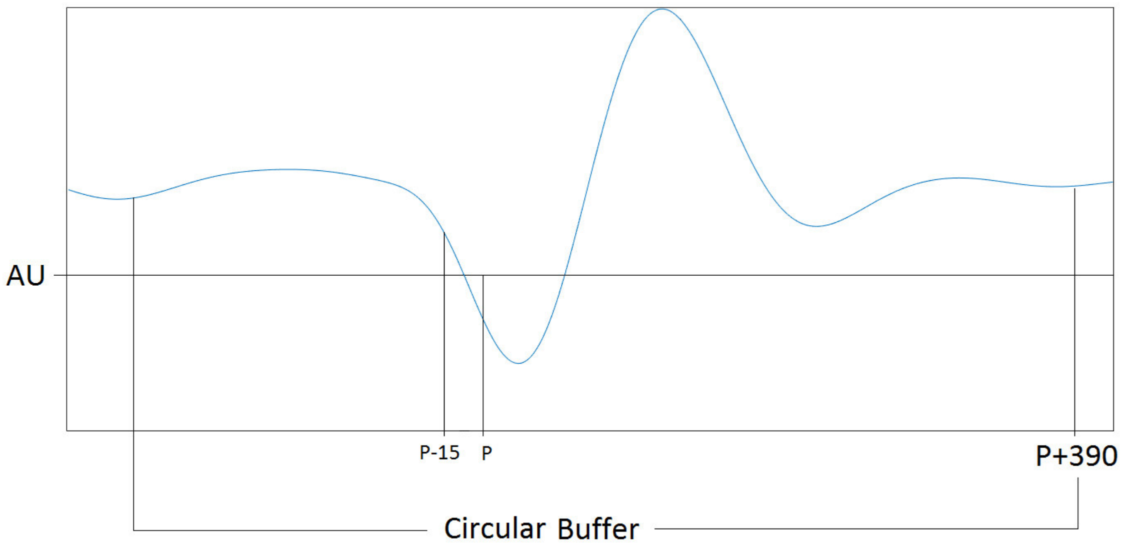


Fig. 7 Example of the application of the blink detection algorithm

Once a sample has been classified as short or long, a sequence of blinks needs to be used to control IoT devices. Commands available to navigate the control menu were represented in blink combinations (see Table 1). Short blinks are represented by a period (.) and long blinks are represented with dashes (-).

Table 1 Commands formed as a combination of blinks.

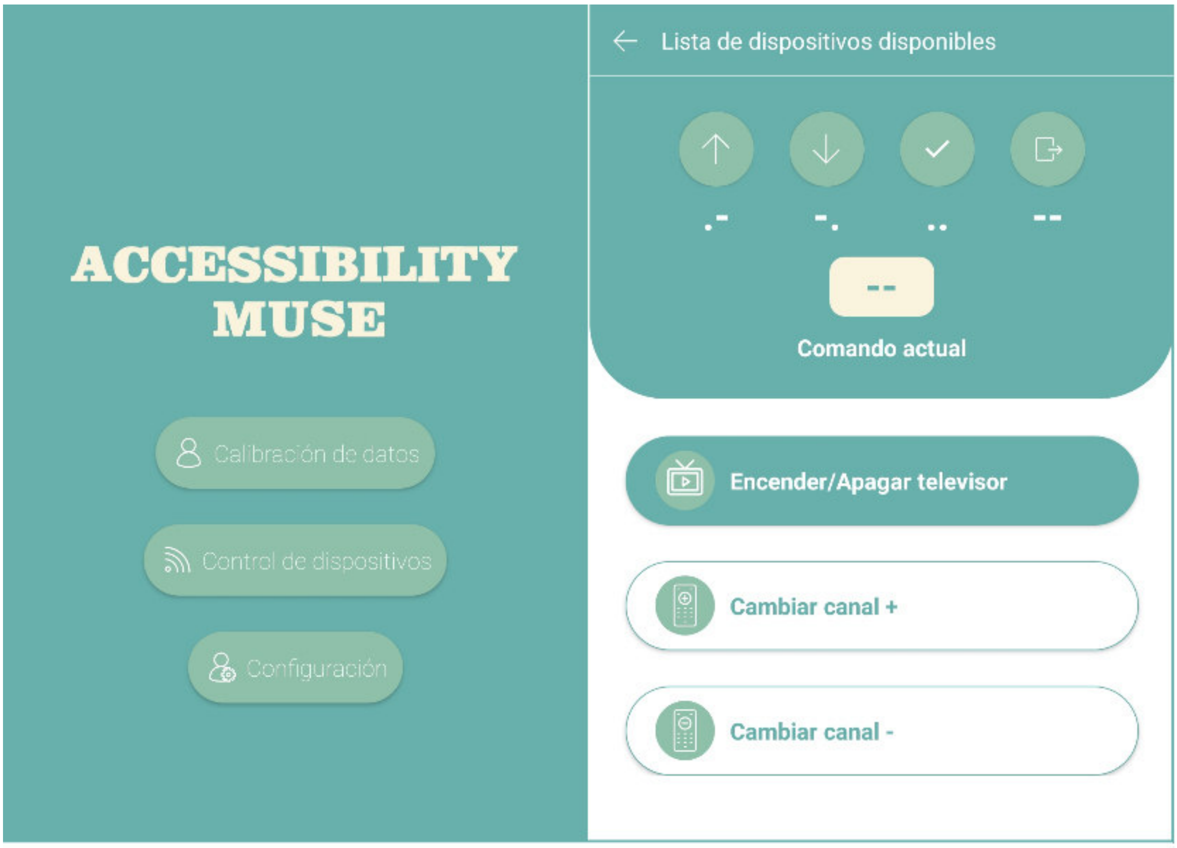
Command	**Description**	**Blink Combination**
Activate / Deactivate	Activates or deactivates the system	Long Long (--)
Up	Select the top option in the menu	Short Long (-)
Down	Selects the lower option in the menu	Long Short (-.)
OK	Executes the selected option	Short Short (..)

2.7. Mobile Application

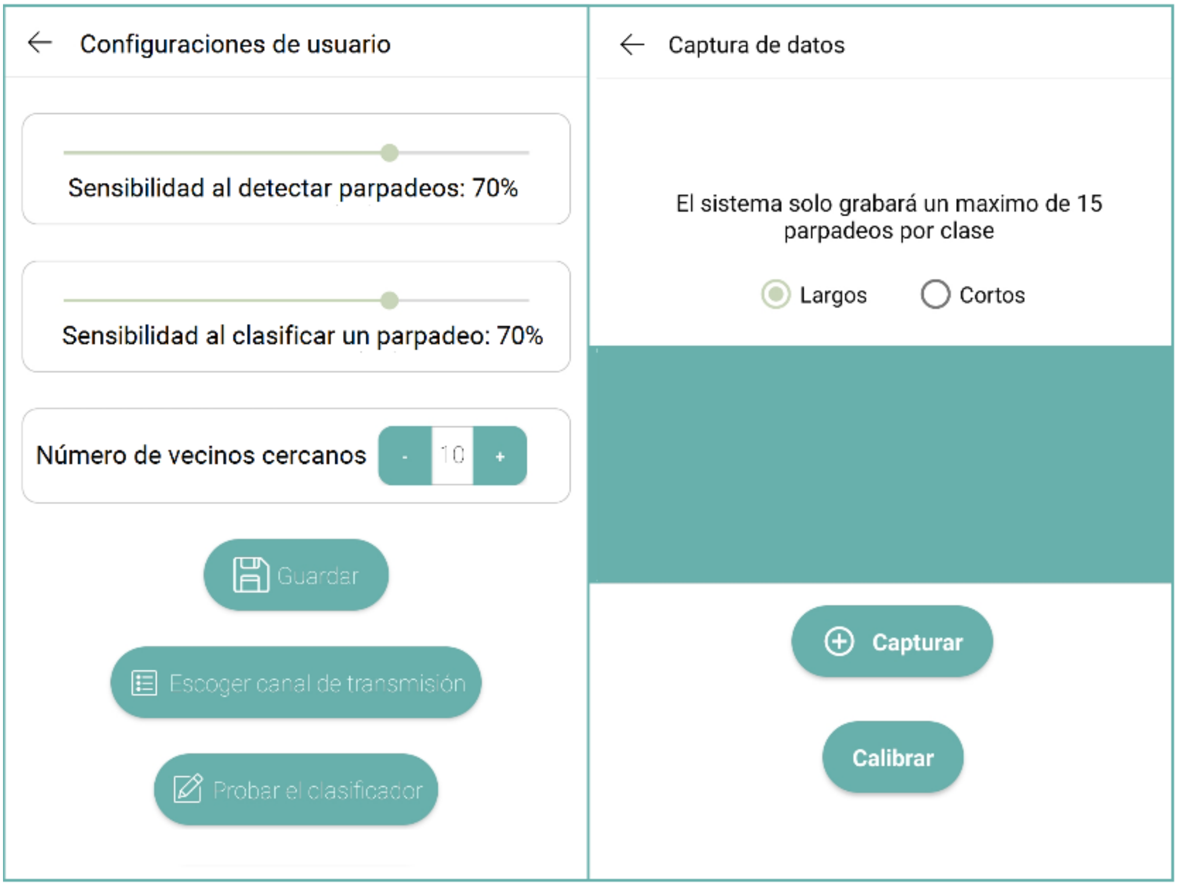
The BCI developed in this work was deployed on a mobile device. For the control of IoT devices, a menu was created showing the available commands and control functions (Fig. 8A). The application also has a module to configure some hyperparameters the k-NN classifier (Fig. 8B). For instance, this module allows us to configure the number k o nearest neighbors and the threshold U for the majority voting of the k-NN algorithm. Whether or not an EEG is sent to the classifier depends on the value of the minimum of amplitude AU. The value of AU is configurable because the amplitude of the signal can vary from one user to another (Fig. 8C).

The classifier works, by default, with a dataset of 15 blinks per class and each sample has a duration of 2 seconds. As the statistical properties of the blink samples can vary from person to person, the application has a calibration section shown in Fig. 8D. This module allows a user to record her/his EEGs for short and long blinks. The user can record from 1 to 15 new samples for each type of blink defined in this work before saving the new dataset.

As the Muse-Headband is a commercial device, it has many models on the market. Two characteristics of the headband that are fundamental for the development of the proposed BCI are the number of channels and sampling amplitudes that could vary according to the model [46]. For this reason, a module for channel configuration and for setting the minimum and maximum sampling rates were also integrated in the mobile application (Fig. 8C).



(A) (B)



(C) (D)

Fig. 8 Main screens of the mobile application.

Version control allows us to keep a backup and track changes that have been made to the code. For this work, GitHub was used as a repository for code versioning. The GitHub project code is available in [47].

Firebase App Distribution was used for the distribution of the application updates. This tool makes easy the task of distributing applications to trusted testers. Version v2.1.3 of the application developed in this work could be accessed through [48].

3. RESULTS

3.1. Classifier performance

Computing the actual accuracy of the classifier developed in this work would require testing it on all the people who could use the system. Since this population is too large, a small set of people was used for the tests. From this group of people, samples of short and long blinks, and no blinks were obtained to build a dataset containing 1500 EEGs. With this size of dataset and assuming a confidence level, the margin of error was calculated to obtain thus a confidence interval for the accuracy of the classifier developed. The margin of error was calculated using Hoeffding's inequality, which provides a universal upper bound for the probability that the sum of random variables deviates a certain amount (i.e., margin of error) from its expected value [49]. Hoeffding's inequality is described in (1), where ϵ is the margin of error, N is the sample size and P is the confidence level. The margin of error corresponding to N = 1500 and P = 0.95 is ϵ = 4.20%.

$$1 - 2e^{-2\epsilon^2 N} = P \quad (1)$$

The acquired dataset was processed by the system developed in this work. Using a confusion matrix (Table 2), the performance of the designed classifier can be visualized. For this test, we used k = 5 (number of nearest neighbors) and a probability threshold of 0.70.

Table 2 Confusion matrix of the classifier developed.

Predicted value \ True value	Short Blink	Long Blink	No Blink	
Short blink	493 32.87%	2 0.13%	5 0.33%	98.60% 1.40%
Long blink	0 0%	500 33.33%	0 0%	100.00% 0.00%
No Blink	0 0%	0 0%	500 33.33%	100.00% 0.00%
	100% 0.00%	99.60% 0.40%	99.01% 0.99%	99.53% 0.47%

Accuracy, precision and sensitivity: Accuracy (Acc) refers to the percentage of correct predictions. Precision (Pr) is the percentage of correct positive predictions. The sensitivity (TP) represents the proportion between the positive cases well classified divided by the total number of positive cases. Taking into account (2), (3) and (4), we have that the accuracy of the classifier proposed is 99.53%, its precisions are greater than 99% and its sensitities are greater than 98%. This means that model is both accurate and exact. Furthermore, since the accuracy is 99.53% and taking into account the calculated margin of error of 4.20%, the 95% confidence interval for the accuracy is between 95.33% and 100%.

$$\mathrm{Acc} = \frac{true\ positives + true\ negatives}{total} \quad (2)$$

$$\mathrm{P} = \frac{true\ positives}{true\ positives + false\ positives} \quad (3)$$

$$\mathrm{TP} = \frac{true\ positives}{true\ positives + false\ negatives} \quad (4)$$

3.2. Usability test

To start the usability tests, the threshold to detect blinks was set to 70%, the threshold to classify a blink was set to 70%, the number of nearest neighbors was set to 5, the reception channel was set to 4, the maximum frequency was set at 975Hz and the minimum frequency at 725Hz. Before starting each of the usability tests, the operation of the system was explained to each user. Additionally, each user was given 5 minutes to practice with the system before evaluating its usability.

To verify the operation of the mobile application developed, 2 tasks were proposed. The first task consisted of the participants performing the following steps: (1) activate the system, (2) turn on a television and (3) deactivate the system. The second task consisted of the following steps: (1) turn on the system, (2) turn on a television, (3) change the transmission channel of the television, (4) turn off the television, and (5) deactivate the system.

The column “Average Test Duration” of Tables 3 and 4 refers to the average time it took a subject to complete 5 repetitions of the corresponding task. The time was recorded from the detection of the first blink to the detection of the last blink. The column “Sent Commands” of these tables refers to the total number of combinations of two blinks sent by the test subject, regardless of whether she/he wanted to send or not that instruction. The column “Incorrect Commands Sent” refers to the total number of combinations of two blinks that the subject did not intend to send.

Table 3 Results of the first task**.**

Subject	Average tests duration [s]	Sent commands	Incorrect commands sent
1	22,74	17	1
2	22,21	17	1

3	22,52	15	0
4	22,68	17	2
5	24,59	15	0

Table 4 Results of the second task.

Subject	Average tests duration [s]	Sent Commands	Incorrect commands sent
1	59,46	42	4
2	60,82	37	1
3	56,29	35	0
4	58,41	39	3
5	57,37	37	1

The results presented in Tables 3 and 4 shows that it was necessary to send, on average, at least 4 commands in an average of 22.95 seconds to complete the first task. To complete the second task, it is necessary to send, on average, at least 8 commands in an average time of 58.46 seconds. The probability of sending an incorrect blink combination in the first task was 4.93% and in the second task it was 4.73%. These probabilities were estimated as the ratio between the number of incorrect commands and the total number of commands that were sent. We can see that there is a reduction in the probability of sending the wrong blink combination. This reduction occurred since, with the time of use of the system (for the first task), the user can more easily remember the combination of blinks that she/he has to send to complete successfully the second task.

4. CONCLUSIONS

In this work, a BCI that uses EEG signals from eye blinks to control IoT devices has been developed. This BCI is especially useful for people with motor disabilities to control IoT devices. The proposed BCI uses combinations of long and short blinks to form commands to control IoT devices. These blinks are identified by the k-NN classifier, which is a very simple and computationally inexpensive algorithm. In addition, the BCI developed includes a mobile application. This mobile application allows the portability of the system developed. The communication between the mobile application and the IoT devices was done through a mosquitto server hosted in the cloud, allowing the user to control any device subscribed to this server. To evaluate the performance of the classifier developed, a test was carried out using 1500 EEGs taken from 10 users. In the classifier tests, the accuracy obtained was 99.53% with a margin of error of 4.2%, with a confidence level of 95%. Additionally, the results of the first usability test, where users had to send an average of 16 combinations of two blinks, showed that the probability of sending an incorrect combination of two blinks was 4.93%. In the second usability test, where users had to send an average of 38 combinations of two blinks, this probability decreased to 4.73%. These results demonstrate that the time of use of the system proposed contributes to reduce the probability of sending a wrong combination of eye blinks. This occurs because with time of use, the user more easily remembers the combination of blinks to send. Future work includes testing new classifiers that can distinguish between voluntary and involuntary blinks.

BIBLIOGRAPHIC REFERENCES

[1] E.-J. H. Claudia Zickler, Valentina Di Donna, Vera Kaiser, Abdul Al-Khodairy, Sonja Kleih, Andrea Kübler, Massimiliano Malavasi, Donatella Mattia, Simona Mongardi, Christa Neuper, Martin Rohm, Rüdiger Rupp, Pit Staiger-Sälzer, "BCI Applications for People with Disabilities: Defining User Needs and User Requirements," vol. 25, pp. 185–189, 2009.

[2] J. R. Wolpaw, "Brain – Computer Interfaces : Something New under the Sun Oxford Scholarship Online Brain – Computer Interfaces : Something New under the Sun Brain – Computer Interfaces : Something New under the Sun," no. September, pp. 2015–2016, 2016, doi: 10.1093/acprof.

[3] D. P. Bhemjibhaih, G. D. Sanjay, V. Sreejith, and B. Prakash, "Brain-computer interface based home automation system for paralysed people," *2018 IEEE Recent Adv. Intell. Comput. Syst. RAICS 2018*, pp. 230–233, 2019, doi: 10.1109/RAICS.2018.8635060.

[4] S. Poveda Zavala, J. L. Bayas Leon, A. Ulloa, J. Sulca, J. L. Murillo López, and S. G. Yoo, "Brain Computer Interface Applicationfor People with Movement Disabilities," in *Human Centered Computing*, 2013, vol. 56, no. 5, pp. 35–47, doi: 10.1007/978-3-030-15127-0_4.

[5] K. Shi, N. Gao, Q. Li, and O. Bai, "A P300 brain-computer interface design for virtual remote control system," *2017 3rd IEEE Int. Conf. Control Sci. Syst. Eng.*, pp. 326–329, 2017, doi: 10.1109/CCSSE.2017.8087950.

[6] K. LaFleur, K. Cassady, A. Doud, K. Shades, E. Rogin, and B. He, "Quadcopter control in three-dimensional space using a noninvasive motor imagery-based brain–computer interface," *J. Neural Eng.*, vol. 10, no. 4, p. 46003, 2013.

[7] B. H. Kim, M. Kim, and S. Jo, "Quadcopter flight control using a low-cost hybrid interface with EEG-based classification and eye tracking," *Comput. Biol. Med.*, vol. 51, pp. 82–92, 2014, doi: 10.1016/j.compbiomed.2014.04.020.

[8] J. R. Wolpaw, D. J. McFarland, G. W. Neat, and C. A. Forneris, "An EEG-based brain-computer interface for cursor control," *Electroencephalogr. Clin. Neurophysiol.*, vol. 78, no. 3, pp. 252–259, 1991, doi: 10.1016/0013-4694(91)90040-B.

[9] J. R. Wolpaw and D. J. McFarland, "Multichannel EEG-based brain-computer communication," *Electroencephalogr. Clin. Neurophysiol.*, vol. 90, no. 6, pp. 444–449, 1994, doi: 10.1016/0013-4694(94)90135-X.

[10] L. Citi, R. Poli, C. Cinel, and F. Sepulveda, "P300-based BCI mouse with

genetically-optimized analogue control," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 16, no. 1, pp. 51–61, 2008, doi: 10.1109/TNSRE.2007.913184.

[11] Y. Li *et al.*, "An EEG-based BCI system for 2-D cursor control by combining Mu/Beta rhythm and P300 potential," *IEEE Trans. Biomed. Eng.*, vol. 57, no. 10 PART 1, pp. 2495–2505, 2010, doi: 10.1109/TBME.2010.2055564.

[12] B. Z. Allison, C. Brunner, C. Altstätter, I. C. Wagner, S. Grissmann, and C. Neuper, "A hybrid ERD/SSVEP BCI for continuous simultaneous two dimensional cursor control," *J. Neurosci. Methods*, vol. 209, no. 2, pp. 299–307, 2012, doi: 10.1016/j.jneumeth.2012.06.022.

[13] L. Bi, J. Lian, K. Jie, R. Lai, and Y. Liu, "A speed and direction-based cursor control system with P300 and SSVEP," *Biomed. Signal Process. Control*, vol. 14, no. 1, pp. 126–133, 2014, doi: 10.1016/j.bspc.2014.07.009.

[14] D. J. Krusienski, E. W. Sellers, D. J. McFarland, T. M. Vaughan, and J. R. Wolpaw, "Toward enhanced P300 speller performance," *J. Neurosci. Methods*, vol. 167, no. 1, pp. 15–21, 2008, doi: 10.1016/j.jneumeth.2007.07.017.

[15] D. J. Krusienski *et al.*, "A comparison of classification techniques for the P300 Speller," *J. Neural Eng.*, vol. 3, no. 4, 2006, doi: 10.1088/1741-2560/3/4/007.

[16] G. Edlinger, C. Holzner, C. Groenegress, C. Guger, and M. Slater, "Goal-oriented control with brain-computer interface," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 5638 LNAI, pp. 732–740, 2009, doi: 10.1007/978-3-642-02812-0_83.

[17] R. Fazel-Rezai and K. Abhari, "A region-based P300 speller for brain-computer interface," *Can. J. Electr. Comput. Eng.*, vol. 34, no. 3, pp. 81–85, 2009, doi: 10.1109/CJECE.2009.5443854.

[18] G. Townsend *et al.*, "A novel P300-based brain-computer interface stimulus presentation paradigm: Moving beyond rows and columns," *Clin. Neurophysiol.*, vol. 121, no. 7, pp. 1109–1120, 2010, doi: 10.1016/j.clinph.2010.01.030.

[19] J. R. Wolpaw, N. Birbaumer, D. J. McFarland, G. Pfurtscheller, and T. M. Vaughan, "Brain–computer interfaces for communication and control," *Clin. Neurophysiol.*, vol. 113, no. 6, pp. 767–791, 2002.

[20] G. R. Müller-Putz, R. Scherer, G. Pfurtscheller, and R. Rupp, "EEG-based neuroprosthesis control: A step towards clinical practice," *Neurosci. Lett.*, vol. 382, no. 1–2, pp. 169–174, 2005, doi: 10.1016/j.neulet.2005.03.021.

[21] F. Piccione *et al.*, "P300-based brain computer interface: Reliability and performance in healthy and paralysed participants," *Clin. Neurophysiol.*, vol. 117,

no. 3, pp. 531–537, 2006, doi: 10.1016/j.clinph.2005.07.024.

[22] I. Iturrate, J. M. Antelis, A. Kübler, and J. Minguez, “A noninvasive brain-actuated wheelchair based on a P300 neurophysiological protocol and automated navigation,” *IEEE Trans. Robot.*, vol. 25, no. 3, pp. 614–627, 2009, doi: 10.1109/TRO.2009.2020347.

[23] A. R. Murguialday *et al.*, “Brain-computer interface for a prosthetic hand using local machine control and haptic feedback,” *2007 IEEE 10th Int. Conf. Rehabil. Robot. ICORR’07*, vol. 00, no. c, pp. 609–613, 2007, doi: 10.1109/ICORR.2007.4428487.

[24] G. Pfurtscheller, T. Solis-Escalante, R. Ortner, P. Linortner, and G. R. Muller-Putz, “Self-paced operation of an SSVEP-based orthosis with and without an imagery-based ‘brain switch’: A feasibility study towards a hybrid BCI,” *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 18, no. 4, pp. 409–414, 2010, doi: 10.1109/TNSRE.2010.2040837.

[25] K. K. Ang *et al.*, “A clinical study of motor imagery-based brain-computer interface for Upper Limb Robotic Rehabilitation,” pp. 5981–5984, 2009.

[26] M. Sarac, E. Koyas, A. Erdogan, M. Cetin, and V. Patoglu, “Brain Computer Interface based robotic rehabilitation with online modification of task speed,” *IEEE Int. Conf. Rehabil. Robot.*, 2013, doi: 10.1109/ICORR.2013.6650423.

[27] C. J. Bell, P. Shenoy, R. Chalodhorn, and R. P. N. Rao, “Control of a humanoid robot by a noninvasive brain-computer interface in humans,” *J. Neural Eng.*, vol. 5, no. 2, pp. 214–220, 2008, doi: 10.1088/1741-2560/5/2/012.

[28] T. Lüth, D. Ojdanić, O. Friman, O. Prenzel, and A. Gräser, “Low level control in a semi-autonomous rehabilitation robotic system via a brain-computer interface,” *2007 IEEE 10th Int. Conf. Rehabil. Robot. ICORR’07*, vol. 00, no. c, pp. 721–728, 2007, doi: 10.1109/ICORR.2007.4428505.

[29] B. S. Baxter, A. Decker, and B. He, “Noninvasive control of a robotic arm in multiple dimensions using scalp electroencephalogram,” *Int. IEEE/EMBS Conf. Neural Eng. NER*, pp. 45–47, 2013, doi: 10.1109/NER.2013.6695867.

[30] U. Hoffmann, J. M. Vesin, T. Ebrahimi, and K. Diserens, “An efficient P300-based brain-computer interface for disabled subjects,” *J. Neurosci. Methods*, vol. 167, no. 1, pp. 115–125, 2008, doi: 10.1016/j.jneumeth.2007.03.005.

[31] D. G. Mohan *et al.*, “A comprehensive review of EEG-based brain-computer interface paradigms,” *Mater. Today Proc.*, vol. 27, no. xxxx, pp. 0–31, 2019, doi: 10.1080/14484846.2018.1432089.

[32] T. Dhanush, B. Aswin Ramnath, T. Sree Sharmila, and J. S. Jennifer, “Determining

Response Credibility by Blink Count," *Proc. 2018 Int. Conf. Recent Trends Adv. Comput. ICRTAC-CPS 2018*, pp. 143–148, 2019, doi: 10.1109/ICRTAC.2018.8679253.

[33] M. Boselli, L. Parrino, A. Smerieri, and M. G. Terzano, "Effect of Age on EEG Arousals in Normal Sleep," *Sleep*, vol. 21, no. 4, pp. 351–357, 1998, doi: 10.1093/sleep/21.4.361.

[34] J. R. Taylor, J. D. Elsworth, M. S. Lawrence, J. R. Sladek, R. H. Roth, and D. E. Redmond, "Spontaneous Blink Rates Correlate with Dopamine Levels in the Caudate Nucleus of MPTP-Treated Monkeys," vol. 220, pp. 214–220, 1999.

[35] A. Subasi and M. I. Gursoy, "EEG signal classification using PCA, ICA, LDA and support vector machines," *Expert Syst. Appl.*, vol. 37, no. 12, pp. 8659–8666, 2010, doi: 10.1016/j.eswa.2010.06.065.

[36] S. Poveda Zavala, K. Ortíz Chicaiza, J. Luis Murillo López, J. Cerezo Ramírez, and S. Guun Yoo, "EEG Signal Processing Model for Eye Blink Detection," *J. Eng. Appl. Sci.*, vol. 15, no. 7, pp. 1671–1675, 2020, doi: 10.36478/jeasci.2020.1671.1675.

[37] S. Poveda Zavala, K. Ortíz Chicaiza, J. Luis Murillo López, J. Sulca, and S. Guun Yoo, "BCI Based Home Automation using User Controlled Blinks," *J. Eng. Appl. Sci.*, vol. 15, no. 6, pp. 1377–1384, 2020, doi: 10.36478/jeasci.2020.1377.1384.

[38] F. L. Colino, "Choosing MUSE : Validation of a Low-Cost , Portable EEG System for ERP Research," vol. 11, no. March, pp. 1–10, 2017, doi: 10.3389/fnins.2017.00109.

[39] D. Java, "Google se hace móvil.," pp. 50–53.

[40] S. Dhanabal and S. Chandramathi, "A Review of various k-Nearest Neighbor Query Processing Techniques," *Int. J. Comput. Appl.*, vol. 31, no. 7, pp. 14–22, 2011, [Online]. Available: http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:A+Review+of+various+k-Nearest+Neighbor+Query+Processing+Techniques#0.

[41] M. Müller, "Chapter 4: Dynamic Time Warping," *Inf. Retr. Music Motion*, pp. 69–84, 2007.

[42] P. Elektrotechniczny and W. Sa, "Processing and spectral analysis of the raw EEG signal from the MindWave Processing and spectral analysis of the raw EEG signal from the MindWave," no. February, 2014, doi: 10.12915/pe.2014.02.44.

[43] "Upload files · Kelvin9811/EEG-Blink-dataset." https://github.com/Kelvin9811/EEG-Blink-dataset/upload/main (accessed May 20, 2021).

[44] R. Of, T. H. E. Committeeon, M. Of, C. Examination, I. City, and T. Committee,

"Report of the committee on methods of clinical examination in electroencephalography," 1957.

[45] G. Hinton and S. Roweis, "Stochastic neighbor embedding," *Adv. Neural Inf. Process. Syst.*, 2003.

[46] InteraXon, "Muse 2 - Technical Specifications," Toronto, Ontario, Canada. [Online]. Available: https://drive.google.com/file/d/1kHLUiVAXODt_jhK_oVtpy3LCqa_nkKcZ/view.

[47] "Kelvin9811/Muse-Accessibility." https://github.com/Kelvin9811/Muse-Accessibility (accessed May 27, 2021).

[48] "Muse-Accessibility – App Distribution – Firebase console." https://appdistribution.firebase.dev/i/fda883ab22bc1648 (accessed May 27, 2021).

[49] "Institute of Mathematical Statistics is collaborating with JSTOR to digitize, preserve, and extend access to The Annals of Statistics. ® www.jstor.org."

ANNEXES

Annexed I Database used for classifier efficiency tests.

- https://github.com/Kelvin9811/EEG-Blink-dataset/upload/main

Annexed II Tool list.

- https://epnecuador-my.sharepoint.com/:f:/g/personal/kelvin_ortiz_epn_edu_ec/EiHQyJ-AnV9GpebAQR50cNABpgVMKbBbYmjyG3-Zudg1sA?e=eeJQZy

Annexed III User manual.

- https://epnecuador-my.sharepoint.com/:f:/g/personal/kelvin_ortiz_epn_edu_ec/ElEueUoSgvlFun42kK4XyfUBHFRL6P0PneffsUcototRuw?e=INnCoh

Annexed IV Installation manual.

- https://epnecuador-my.sharepoint.com/:f:/g/personal/kelvin_ortiz_epn_edu_ec/ElEueUoSgvlFun42kK4XyfUBHFRL6P0PneffsUcototRuw?e=INnCoh

Annexed V Access to the latest version of the mobile application.

- https://epnecuador-my.sharepoint.com/:f:/g/personal/kelvin_ortiz_epn_edu_ec/EtNi0P2rQ8RGtUmjZWoBw_EBJ4abJD1CrkVgm5sQiSE03w?e=sHtsdH