

UNIVERSIDAD MIGUEL HERNÁNDEZ DE ELCHE
ESCUELA POLITÉCNICA SUPERIOR DE ELCHE
GRADO EN INGENIERÍA ELECTRÓNICA Y
AUTOMÁTICA INDUSTRIAL



“PREDICCIÓN EN TIEMPO REAL DE LA
PARADA DE UN USUARIO ANTE UN
OBSTÁCULO MEDIANTE EEG”

TRABAJO DE FIN DE GRADO

Diciembre - 2021

AUTOR: Alejandro Juan Moreno Calderón

DIRECTOR/ES: Eduardo Iáñez Martínez



RESUMEN

El objetivo del presente trabajo es el estudio a tiempo real y la obtención de señales cerebrales a partir de un obstáculo inesperado mientras el usuario se encuentra en una marcha normal a una velocidad constante.

Para ello se ha llevado a cabo un experimento con 5 usuarios, mientras realizaban la acción previamente citada. Para la obtención de estas señales se han utilizado tanto IMUs por la parte inferior del cuerpo como electrodos en la cabeza del usuario, además de una cinta de correr para mantener una velocidad constante y la simulación de un obstáculo mediante un puntero láser.





AGRADECIMIENTOS

En primer lugar, me gustaría agradecer a mi tutor, Eduardo Láñez Martínez, la oportunidad de realizar este proyecto, con el cual he aprendido muchas cosas y de la que voy a guardar un gran recuerdo, y la dedicación que me ha prestado durante todo este proceso.

En segundo lugar, me gustaría agradecer a mis voluntarios, más conocidos como mis amigos, que han estado en todo momento y me han ayudado en todo lo pedido.

Por último, a mi familia, que no ha dejado que me desanimara y me ha apoyado a seguir en todo momento.





ÍNDICE GENERAL

1. INTRODUCCIÓN
 - 1.1 Motivación
 - 1.2 Objetivos
 - 1.3 Organización del proyecto
2. Estado del arte
 - 2.1 Generalidades del sistema nervioso
 - 2.2 Electroencefalografía
 - 2.2.1 Registro
 - 2.2.2 Posicionamiento de los electrodos
 - 2.2.3 Potenciales
 - 2.2.3.1 Potencial relacionado con la aparición de un obstáculo
 - 2.2.3.2 Potencial de error
 - 2.2.3.3 Potencial de la intención de detenerse
 - 2.3 Interfaces cerebro-máquina
 - 2.3.1 Adquisición de datos
 - 2.3.2 Preprocesamiento
 - 2.3.3 Extracción de características
 - 2.3.4 Clasificación
 - 2.4 Aplicaciones de las BMI en rehabilitación
3. Material y método
 - 3.1 Material
 - 3.1.1 Equipo de medición EEG
 - 3.1.2 IMUs
 - 3.1.3 Láser y cinta de correr.
 - 3.2 Protocolo experimental para registro
 - 3.2.1 Protocolo experimental offline
 - 3.3 Método automático para la detección de la parada mediante IMUs
 - 3.4 Análisis de la actividad cerebral
 - 3.4.1 Preprocesamiento
 - 3.4.2 Selección de electrodos
 - 3.4.3 Selección de ventanas
 - 3.4.3.1 Selección de ventanas para la clase 1
 - 3.4.3.2 Selección de ventanas para la clase 2
 - 3.4.4 Extracción de características
 - 3.4.5 Clasificadores
 - 3.4.5.1 LDA
 - 3.4.6 Análisis pseudo-online
 - 3.4.7 Métricas
 - 3.5 Utilización arquitectura
 - 3.6 Mejoras realizadas en los códigos
 - 3.7 Protocolo experimental online
4. Resultados y discusión.
 - 4.1 Usuarios
 - 4.2 Valores iniciales de dif_media y muestra
 - 4.3 Comprobación y adaptación código antiguo
 - 4.4 Resultados pruebas offline

- 4.5 Realización pruebas online
- 4.6 Resultados pruebas online

5. Conclusión.

Bibliografía





Índice de figuras

- Figura 1. Potencial de acción.
- Figura 2. Tipos de señales cerebrales.
- Figura 3. Disposición de los 32 electrodos según sistema internacional 10-10.
- Figura 4. Posición electrodos.
- Figura 5: Ejemplo potencial relacionado con la aparición de un obstáculo.
- Figura 6. Sistema BMI
- Figura 7. Casco electrodos.
- Figura 8. Actichamp.
- Figura 9. Equipamiento IMUs.
- Figura 10. Posición IMUs.
- Figura 11. Montaje experimental.
- Figura 12. Tareas (400, 500, 600) y tiempos en segundos del modo offline.
- Figura 13: Usuario en prueba.
- Figura 14. Función laser_vector.
- Figura 15. Findchangepts correcto.
- Figura 16. Findchangepts incorrecto.
- Figura 17: Esquema del modelo empleado.
- Figura 18. Mapa electrodos con la selección.
- Figura 19. Comparación muestra usuarios W55 y W30.
- Figura 20. Representación clase 1 usuario W26.
- Figura 21. Representación clase 2 usuario W26.
- Figura 22. Tareas protocolo Online.
- Figura 23. W26 obtención muestra código nuevo.
- Figura 24. W58 obtención muestra código nuevo.
- Figura 25. Resultados usuario W55.
- Figura 26. Resultados primera repetición W55.
- Figura 27. Resultados segunda repetición W55.

Figura 28. Resultados tercera repetición W55.

Figura 29. Resultados cuarta repetición W55.

Figura 30. Resultados quinta repetición W55.

Figura 31. Resultados sexta repetición W55.

Figura 32. Resultados usuario W58.

Figura 33. Resultados primera repetición W56.

Figura 34. Resultados segunda repetición W56.

Figura 35. Resultados tercera repetición W56.

Figura 36. Resultados cuarta repetición W56.

Figura 37. Resultados IMUs W58.

Figura 38. Resultados IMUs W55.



Índice de tablas

Tabla 1. Número de características.

Tabla 2. Explicación datos resultado código pseudo.

Tabla 3: Promedios dif_media de los 6 usuarios.

Tabla 4. Resultados muestra y dif_media usuarios nuevos.

Tabla 5. Electrodo por usuario nuevo.

Tabla 6. Electrodo por usuario antiguo.

Tabla 7. Configuración utilizada.

Tabla 8. Resultados 9/1 usuarios antiguos y usuarios nuevos.

Tabla 9. Resultados usuarios nuevos offline 5/5.

Tabla 10. Resultados para usuarios antiguos offline 5/5.

Tabla 11. Resultados W26 configuración 7/3.

Tabla 12. Resultados W26 configuración 2/8.

Tabla 13. Resultados comparación códigos.

Tabla 14. Resultados dif_media, muestra y electrodo usuarios modo online.

Tabla 15. Resultados configuración 4+1 modo offline

1. INTRODUCCIÓN

Las enfermedades cerebrovasculares (ECV) representan un problema a nivel de salud mundial, ya que hoy en día constituye la patología neurológica más común y primera causa de discapacidad en la población [12]. Con una cifra de fallecidos muy amplia. También puede provocar discapacidades en un alto porcentaje a la población que lo sufre [10].

En los últimos años los estudios de la utilización cerebro-máquina (BMI, del inglés Brain-Machine Interface) han aumentado, dando así una dimensión mayor a la rehabilitación de personas con accidentes o enfermedades, las cuales han provocado enfermedades a nivel de médula en los sujetos. Esta tecnología permite la rehabilitación mediante exoesqueleto además de la asistencia, ya que los primeros meses con este tipo de lesiones son muy importantes debido a la plasticidad del sistema nervioso, por lo que esta rehabilitación no supondrá un esfuerzo tan grande como el actual para los pacientes.

El uso de estos exoesqueletos viene acompañado de un software que actúe de interfaz cerebro-maquina y que permita la realización de movimientos que sin una ayuda externa podrían resultar muy complejos para los pacientes al comienzo de las terapias aplicadas con esta tecnología.

El sistema BMI se puede definir como un sistema que traduce y procesa las señales emitidas por el cerebro para mandar órdenes a un dispositivo sin la intervención de ninguna otra parte del cuerpo. Así permite la realización de movimientos extras sin ayuda.

En el presente trabajo estudiaremos la obtención de las señales generadas un usuario cuando se decide a detenerse cuando ha detectado un obstáculo para luego continuar la marcha y poder volver a generar otra señal para poder recoger otra señal, esta obtención de señales se hará mediante métodos no invasivos como IMUs o electrodos, y más tarde estas señales se podrán aplicar a la interfaz de un exoesqueleto, posibilitando así un uso en la vida real sin riesgo para el paciente y de forma útil y segura.

Lo que deseamos buscar con este proyecto es la forma en la que el exoesqueleto se pueda detener de forma segura durante las pruebas que se realicen con pacientes reales, para poder así evitar los accidentes.

1.1 Motivación

Actualmente, existe mucha reticencia por parte de la sociedad a los avances tecnológicos, ya que se tiene la sospecha o se cree que un futuro próximo dichas máquinas provocadas por los avances tecnológicos podrían sustituirnos o que una inteligencia artificial pueda superarnos en cualquier instante, sin embargo, el tema que tratamos en este trabajo es enfocado a la salud de las personas y a la mejora de la calidad de vida.

Mediante, este trabajo estamos entrando en un ámbito muy satisfactorio y con mucho camino que recorrer, ya que con estos nuevos estudios se pueden conseguir mayor precisión en las soluciones para las personas, por lo que la tecnología juega un papel muy importante en todo esto.

Realizar actividades diarias como correr, levantarse, agacharse... Que nosotros consideramos básicas y triviales resultan un gran esfuerzo o simplemente un deseo para muchas personas afectadas por este tipo de enfermedades, por lo que es muy gratificante adentrarse en esta rama de la ingeniería más enfocada en ayudar a estas personas a tener una más que mejor calidad de vida y poder ayudarle a afrontar el día a día de una persona cotidiana sin mayor esfuerzo.

1.2 Objetivos.

El objetivo principal del presente trabajo es detectar la aparición de un obstáculo inesperado para el usuario durante la marcha normal mediante señales electroencefalográficas (EEG). De esta forma se podría estudiar y analizar las señales mandadas por el cerebro durante la ejecución de la señal de parada que emite el cerebro, para implementarla en códigos futuros y así poder obtener nuevos avances tecnológicos.

Se pretende así ir aproximándonos a una implementación en la vida real cada vez más próxima. Todas estas señales se obtienen con técnicas no invasivas, tanto en cabeza como en piernas mediante electrodos y IMUs, se pretende encontrar principalmente la actividad cerebral tanto en la zona occipital, debido al estímulo visual, como en la zona inferior del cuerpo (motora).

El principal objetivo es poder llegar a aplicar este código en tiempo real, así podremos darle la oportunidad al paciente a poder detenerse cuando durante su caminata se enfrente a un obstáculo inesperado y el pretenda detenerse el desee y no dependa de un agente externo como pueda ser una persona para su ayuda.

1.3 Organización del proyecto.

El presente trabajo se divide en 5 capítulos los cuáles tratarán los siguientes aspectos:

- Capítulo 1: En este primer capítulo se trata una breve introducción al proyecto, la organización y los objetivos de este.
- Capítulo 2: Contiene una visión general de los conceptos más importantes que trataremos en un futuro y explicaciones de los diferentes conceptos que se verán en este TFG. Se explicará cómo se trabaja con una BMI y los exoesqueletos además de la interfaz cerebro-máquina.
- Capítulo 3: Trataremos los materiales usados durante la realización de las pruebas y el proceso de obtención de los resultados. Se explicará la obtención de los datos y de los algoritmos empleados durante el proceso.
- Capítulo 4: En el cuarto punto comentaremos los resultados obtenidos para cada usuario. Se diferenciarán los diferentes métodos usados, tanto offline como online.
- Capítulo 5: En el capítulo final realizaremos un comentario general sobre todo el trabajo y llegaremos a una conclusión, además propondremos futuras investigaciones a partir de los resultados que hemos obtenido.



2. Estado del arte

En este segundo capítulo veremos el estado del arte, consistirá en explicar el ámbito de estudio en el cual se encuadra, el proyecto y per se la investigación que hemos llevado a cabo.

Se explicará mediante una breve introducción, las generalidades del sistema nervioso para poder así entender su funcionamiento y las partes importantes que lleva a cabo, además estudiaremos la interfaz cerebro-máquina, también conocido como BMI, el cual veremos sus aplicaciones y funcionamientos además de sus diferentes clasificaciones, usos, y sus etapas en el desarrollo del estudio de esta como puede ser adquisición de datos, preprocesamiento, extracción de características y clasificación de estas señales.

Más tarde continuaremos con las diversas aplicaciones que tienen los sistemas BMI, y el enfoque que nosotros le daremos en nuestras investigaciones y desarrollos, como es la rehabilitación.

Finalizaremos con un estudio y una descripción de una electroencefalografía, ya que gracias a esto es como podemos obtener todos los datos que requieren el estudio, y terminaremos con el estudio de las diferentes ondas cerebrales que podemos encontrarnos en nuestro cerebro y una breve descripción de cada una de ella.

2.1. Generalidades del sistema nervioso.

El sistema nervioso, por lo general, es una red compleja de estructuras, las cuales tienen como misión controlar y regular el funcionamiento de los diferentes órganos del cuerpo humano y sistemas de este. Es el sistema más completo y sofisticado del cuerpo humano. El sistema se divide en 2 grandes subsistemas el sistema nervioso central y el sistema nervioso periférico.

Ambos, para comunicarse con las distintas partes del cuerpo, recurren a uniones sinápticas entre diferentes neuronas, las cuales generan señales que conocemos como potencial de acción.

Un potencial de acción consiste en un cambio repentino, rápido, transitorio y que se propaga en el potencial de la membrana de reposo, producido por un intercambio de iones.

Para que se produzca este potencial es necesario que actúe un estímulo superior a un determinado umbral, en la figura 1 se verá su forma característica.

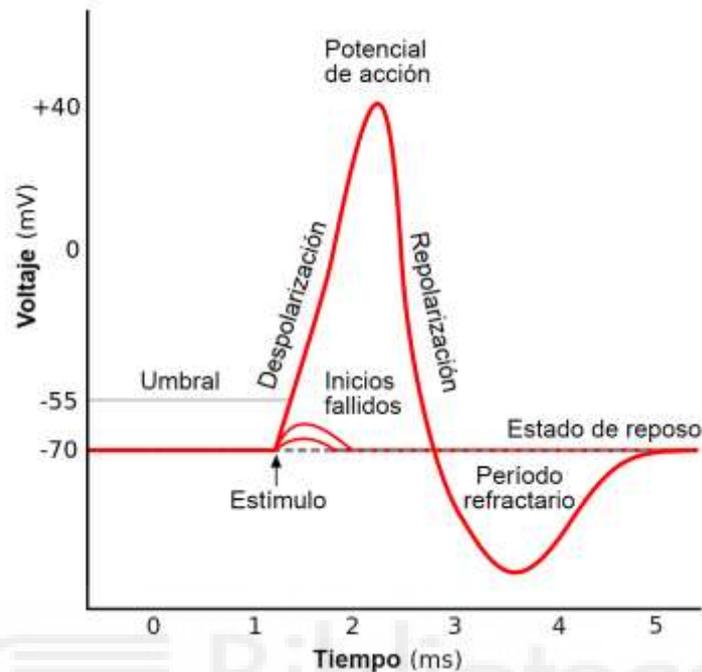


Figura 1. Potencial de acción [3].

2.2 Electroencefalografía

La electroencefalografía es una técnica no invasiva utilizada para poder obtener la actividad eléctrica del cerebro mediante pequeños electrodos sobre el cuero cabelludo, registrando así datos de toda la superficie del cerebro. El registro que obtenemos mediante esta técnica se llama encefalograma, o también conocido como EEG.

El origen de este tipo de prueba se remonta al siglo XX, más específicamente al año 1942, cuando el neurólogo Hans Berger, le realizó este tipo de técnica a su hijo, pero debido a una mala acogida por la gran mayoría de la comunidad científica, su progreso se vio cortado durante una década, para más tarde seguir con dichas técnicas, las cuales ampliaron y veremos a continuación.

Las ondas de EEG, se pueden clasificar en diferentes tipos, la clasificación que realizamos en este trabajo es por rangos de frecuencia y su implicación, se pueden dividir en los siguientes grupos:

-**Ondas Delta:** Estas ondas, oscilan entre 0.5 y 4 Hz, son las que poseen una mayor amplitud y se suelen encontrar en situaciones de sueño profundo.

-**Ondas Theta:** Las ondas theta están comprendidas entre las frecuencias de 4 y 8 Hz, estas ondas se suelen dar en estados de calma y/o relajación.

-**Ondas Alfa:** Las ondas alfa, son las situadas entre las frecuencias de 8 y 12Hz, su origen es sobre todo producido en el lóbulo occipital durante periodos de relajación, pero no tan profundas como las ondas theta.

-**Ondas Beta:** Comprende las frecuencias que se mueven entre 12 a 35Hz, son un tipo de ondas que se producen a niveles altos de concentración, como por ejemplo a la hora de una decisión o la resolución de un problema.

-**Ondas Gamma:** Este tipo de ondas se mueve entre 35Hz y frecuencias superiores, este tipo de ondas se dan en acciones que requieren un alto procesamiento y concentración.

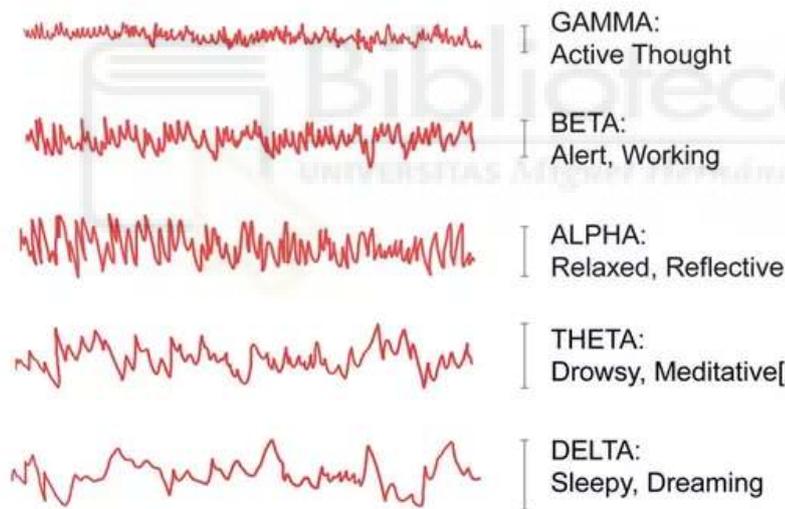


Figura 2. Rangos frecuencias señales EEG [4].

2.2.1 Registro

-Invasivas: En las técnicas invasivas, el dispositivo que nos ayuda a obtener las señales se instala dentro del cerebro de la persona, este método es el más peligroso de los dos, pero el cual nos da unos datos más exactos y específicos, ya que la relación señal-ruido es la más baja.

-No invasivos: Este es el método más seguro, ya que no requiere la instalación de ningún dispositivo. Si no, ya solo con un gorro de electrodos, como hacemos

en nuestras pruebas podemos obtener todos los datos que queramos, el problema de este método es que es menos exacto que el anterior y puede generar mucho ruido, además de ser mucho más económico.

2.2.2 Posicionamiento de los electrodos

Para poder obtener una señal precisa de la actividad eléctrica en cada una de las regiones cerebrales a través de la superficie craneal es necesario utilizar una serie de electrodos que captan la diferencia de potencial entre ellos. Se analizarán las señales cerebrales ampliando las diferencias de potencial entre los electrodos receptores.

Estos electrodos se posicionarán de acuerdo con el sistema internacional 10-10, el cual se llama así debido a que los electrodos están espaciados entre el 10% y el 10% de la distancia total entre puntos reconocibles del cráneo.

Los distintos electrodos se combinan, constituyendo los montajes. Existen dos tipos de montajes, el bipolar y el monopolar.

El montaje bipolar recopila la diferencia de voltaje entre dos electrodos colocados en áreas de actividad cerebral, en cambio el montaje monopolar registra la diferencia de potencial entre un electrodo ubicado en una zona cerebral que se desea estudiar y otro colocado en una zona con una actividad escasa, o sin actividad.

Además, dispondremos de cuatro electrodos más que situaremos por el rostro de la persona los cuales usaremos para poder eliminar el ruido que se introduce por esa parte por la persona, se situarán a los laterales de ambos ojos y tanto en zona inferior como superior del ojo izquierdo.

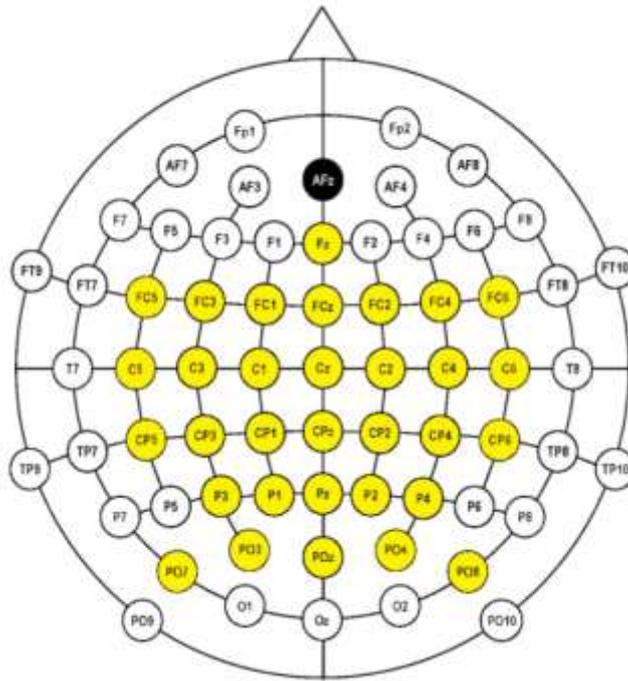


Figura 3: Disposición de los 32 electrodos según sistema internacional 10-10



Figura 4. Posición electrodos.

2.2.3 Potenciales

Un potencial es una de las señales EEG más estudiadas dentro de los campos de la neurología o la psicología. Es la medida de la respuesta cerebral que es el resultado directo de un evento específico sensorial, cognitivo o motor.

Los potenciales pueden ser medidos usando electroencefalografía (EEG). Este tipo de señales tienen una amplitud de onda muy pequeña de entre 1 y 30 microvoltios. El mayor inconveniente que nos encontramos a la hora de clasificar estas señales se debe al alto grado de variabilidad que tienen entre los distintos sujetos. Esto nos obliga a realizar un continuo reentrenamiento de los clasificadores.

Por lo general, cualquier Potencial Relacionado con un Evento (ERP) se encuentra totalmente enmascarado dentro de un ruido de fondo que presenta una mayor amplitud, por lo que es necesario intentar extraer la señal de registro.

2.2.3.1 Potencial relacionado con la aparición de un obstáculo

Uno de los principales objetivos de nuestro trabajo será el analizar la activación cerebral que se genera cuando aparece un obstáculo no esperado por el usuario durante la marcha normal, por ese motivo es por lo que se clasifican dentro de los potenciales relacionados a eventos.

En nuestra investigación pretendemos analizar la señal producida por el cerebro de los diferentes usuarios que se han prestado voluntarios durante la caminata en la cinta. Durante toda esta caminata se va a ir analizando el proceso cerebral de los usuarios y ver en el momento en el que se decide para por el usuario. Estas señales pueden variar tanto positivamente como negativamente, ya que se pueden producir en distintos tiempos.

En el potencial relacionado con los obstáculos podemos encontrar varios potenciales fundamentales que están relacionados con los ERP más importantes como pueden ser la variación negativa contingente (CNV), que suele ir seguido de una respuesta mental o motora.

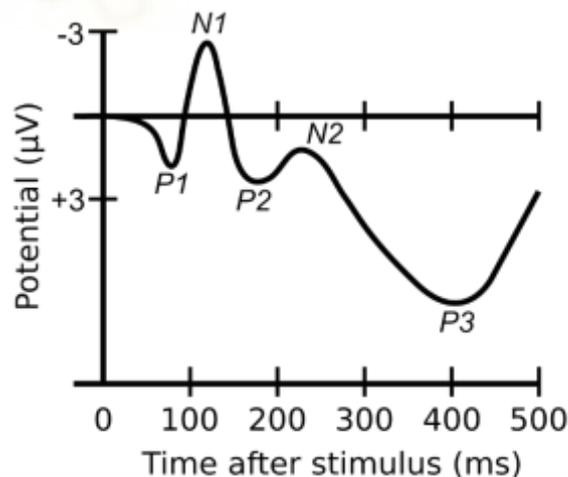


Figura 5: Ejemplo potencial relacionado con la aparición de un obstáculo [5].

2.2.3.2 Potencial de error.

El potencial de error se compone por los potenciales que se generan cuando se produce una estimulación externa, ya sea visual o auditiva. Esta señal se

considera de error ya que puede contener información no deseada para nuestra prueba.

Este potencial se caracteriza por ser un potencial negativo que se sitúa entre los 50 y 100 ms. Es necesario que para cada usuario que participe en las pruebas se calibre este potencial, debido a que a la variabilidad de la señal ERP. [5] Debido a que el usuario está en alarma de que aparezca el obstáculo de un momento a otro es poco probable que podamos apreciar la aparición de este potencial, salvo en algunos casos concretos.

2.2.3.3 Potencial de la intención de detenerse

Dentro de los potenciales relacionados con los eventos nos encontramos también con el potencial relacionado con la parada, este potencial se produce de media entre los 700 y 900 ms después de que se haya producido el estímulo visual y justo antes del movimiento de detección por parte del usuario.

2.3 Interfaz cerebro-máquina.

La interfaz cerebro-máquina, se puede definir como un sistema capaz de poder procesar las ondas cerebrales, y decodificarlas para así poder obtener una serie de comandos de control para que el usuario pueda controlar un dispositivo externo a esto sin la necesidad de tener que mover ningún músculo adicional, esta interfaz es conocida también como BMI y se puede llegar a obtener a partir de EEG.

Las señales que estudiamos son de un gran uso y de una gran ayuda, ya que se pueden aplicar al estudio de la robótica, y más allá a la rehabilitación como veremos en este trabajo, y a todo el mundo de la medicina.

Estas señales se pueden obtener mediante técnicas invasivas, o técnicas no invasivas, que son las cuales veremos más a fondo en este trabajo y las estudiaremos.

Además de todo esto, estas señales se pueden dividir en varios grandes grupos, que son: Adquisición de la señal, preprocesamiento, extracción de características, clasificación de estas señales y aplicación.

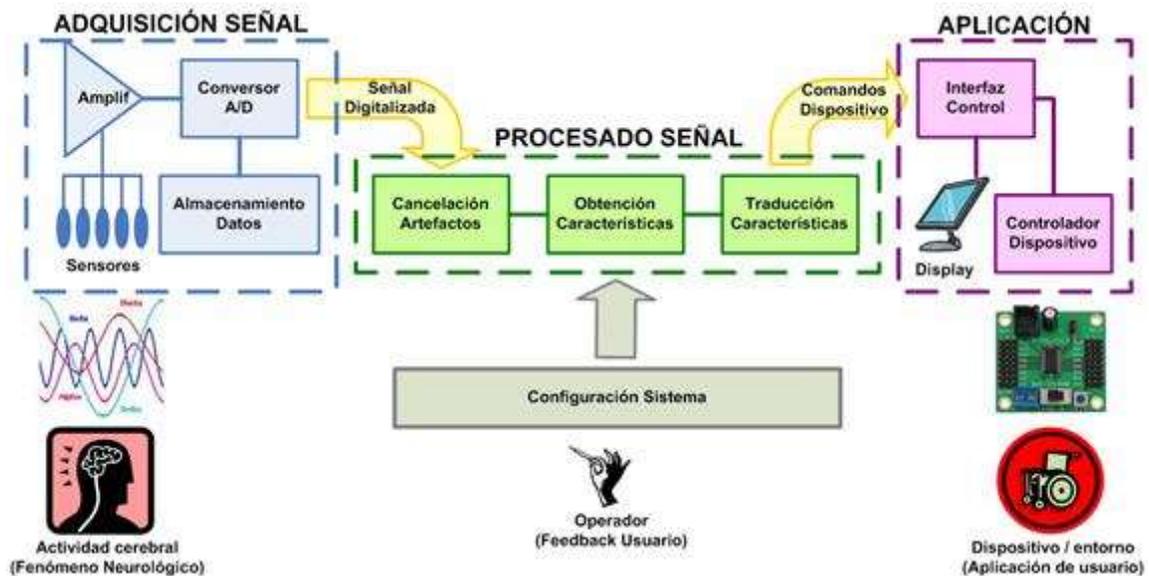


Figura 6. Sistema BMI [6].

2.3.1. Adquisición de datos.

Este será el primer paso por realizar, el cuál consistirá en recoger toda la información ofrecida por el usuario durante la prueba, para después poder usar en nuestros estudios y aplicarle los métodos necesario para la obtención de nuestro objetivo.

En nuestras pruebas, esta adquisición de datos vendrá de dos formas diferentes, una será la obtención de la información motora del usuario, esta obtención se hará mediante sensores colocados en las piernas del usuario llamados IMUs, y después la obtención de la señal cerebral que se hará mediante un gorro de electrodos, el cual nos aportará toda la información necesaria para nuestra prueba.

2.3.2. Preprocesamiento.

Para poder trabajar más adelante con una señal limpia lo cual haga que nuestras pruebas del proyecto sean más eficaces, aplicaremos un preprocesamiento nos ayuda a obtener una señal más limpia y correcta mediante la aplicación de filtros a la señal, como el filtro de Notch, para más tarde evitarnos variaciones por las diferentes frecuencias. Además, aplicamos filtros como el CAR o el Laplaciano, toda esta combinación resulta en la señal que obtenemos y que posteriormente estudiamos.

2.3.3. Extracción de características.

La extracción de características la usaremos para poder realizar una distinción entre las diferentes clases que usaremos en nuestro proceso.

Las características podemos clasificarlas en dos grandes grupos, como son las características frecuencias y las características temporales.

Las características frecuenciales se obtienen como resultado de transformar la señal en el dominio de la frecuencia y tiene como objetivo principal determinar la potencia de cada señal basada en el análisis de la transformada rápida de Fourier (FFT). Este tipo de características presentan una gran ventaja que es su relativa rapidez de cálculo, pero también presentan una serie de desventajas como que las frecuencias utilizadas para descomponer una señal son una función de la frecuencia de muestreo de la sea y del número de frecuencias [7].

Las características temporales las encontramos cuando se extraen las señales EEG una vez se han filtrado y extraído las deseadas, este proceso se realiza en el dominio del tiempo. Existen diversos métodos de cálculo de estas características como los modelos autorregresivos, donde se utilizan los coeficientes del modelo generado como característica. Posee ventajas respecto al frecuencia en su resolución de frecuencia y buenas estimaciones espectrales. En nuestro proyecto usaremos este tipo de características.

La extracción de características deseadas se obtendrá, principalmente, mediante la aplicación de los algoritmos creados para el estudio y obtención de estos datos. Dependiendo de los datos que queramos usar para después poder clasificarlos, usaremos unos u otros.

2.3.4. Clasificadores.

Un clasificador es un método que debemos usar justo después de haber obtenido las características para poder identificarlas y clasificarlas según sus características.

Para realizar este proceso existen multitud de clasificadores disponibles que nos permiten poder realizar este paso. Nosotros hemos realizado el estudio de dos de ellos, el clasificador SVM (Support Vector Machine) y LDA (Lineal Discriminate Análisis).

El sistema de clasificación LDA consiste en un método supervisado de variables cualitativas en el cual, los grupos son conocidos a priori y se van clasificando según las características que tengan, haciendo uso del teorema de Bayes, se estima la probabilidad y se clasificada en una clase u otra.

El método de clasificación SVM, se basaba en sus inicios, en un método de clasificación mediante números binarios, pero ha ido evolucionando hasta el día de hoy convirtiéndose en uno de los mejores clasificadores y más usados dentro del mundo del maching learning.

Finalmente se ha empleado para nuestro estudio el clasificador LDA, ya que los resultados que nos arrojaba eran ampliamente mejores.

Si se implementan de forma correcta, estos sistemas puedes ser capaces de interpretar en tiempo real los datos que van generándose durante la marcha de un individuo.

2.4 Aplicaciones BMI.

Esta tecnología tiene muchas posibles aplicaciones en la vida cotidiana y también tiene mucho potencial de mejora y de descubrir nuevas aplicaciones en el futuro. Una de las más importantes es la aplicada a las enfermedades cerebrovasculares, en las cuales se ha demostrado que si se realiza un tratamiento durante los primeros 6 meses después de haber sufrido un accidente se puede llegar a recuperar considerablemente más rápido la persona, debido a la plasticidad del sistema nervioso.

Es en este punto donde entra la tecnología de las BMI, ya que en los últimos años ha sufrido una gran evolución y se han ido desarrollando hasta el día de hoy. Ahora hay estudios muy enfocados en los robots terapéuticos como los exoesqueletos, los cuales se adaptan a las necesidades de la persona y le ayudan a poder mejorar su condición con sus determinadas características, lo cual hace más fácil el esfuerzo que tiene que realizar la persona justo después de haber sufrido el accidente.

Este exoesqueleto acompaña en los movimientos que quiere realizar el usuario, así reduce su esfuerzo. Este tipo de exoesqueleto tiene mucho control gracias a la tecnología BMI.

En esta investigación desarrollamos una parada de emergencia con seguridad antes la detección de un obstáculo lo cual hará que el usuario que este siendo

ayudado por el exoesqueleto pueda ir libremente sin la necesidad de más ayuda extra.

Además de este estudio existe muchos más como los cambios de velocidad. Este estudio consiste en el que el usuario pueda elegir cuando sea ir más rápido o ir más lento en función de sus condiciones

También nos encontramos un estudio el cual ayuda al paciente a elegir la dirección deseada cuando este quiera cambiarla.

Pero no solo existen aplicaciones BMI en el ámbito de la rehabilitación y la asistencia si no que existen en otros muchos más campos.

[8] Se han desarrollado aplicaciones para el control de un teléfono móvil mediante el uso de este tipo de señales. Este tipo de aplicaciones permiten al usuario controlar las tareas de un teléfono como coger llamadas o mandar un SMS con este método. Este proyecto obtuvo resultados muy satisfactorios con una media de acierto de más del 95%.

También existen proyectos [9] para poder controlar dispositivos domóticos y electrónicos de una vivienda. Consistía en intentar facilitar la vida a las personas de más avanzada en el control del funcionamiento de la casa, como puede ser encender la calefacción o bajar las persianas, los resultados no fueron tan buenos como los obtenidos para la aplicación móvil.



3. Material y métodos.

En este tercer capítulo, veremos cuales han sido los materiales que hemos empleado en todas las pruebas realizadas a los usuarios y los diferentes pasos que hemos seguido.

Además de todo esto, se describirán detenidamente los pasos experimentales para la obtención de los datos y los métodos para el procesamiento y la clasificación de las señales EEG. Para finalizar con una breve explicación del diferente uso de los electrodos y de las posiciones escogidas para la obtención de dichos datos.

3.1. Material.

En este proyecto hemos utilizado diversos componentes para poder obtener los resultados, tales como equipo EEG, el cual nos ayudó a obtener los datos del usuario al instante, una cinta de correr para simular el caminar del usuario, un láser que simula la aparición de un obstáculo y para finalizar un sistema de captura de movimiento para obtener todos los resultados que nos llegará del usuario mediante el movimiento.

3.1.1. Equipos de medición EEG.

El equipo fundamental para obtener los datos principales de nuestro proyecto es el equipo de medición EEG. Esto consiste en medir todos los impulsos eléctricos que se generan en la cabeza durante la marcha, para ello emplearemos diferentes componentes como es el gorro de electrodos, el amplificador Actichamp y el programa Pycorder para ver las señales del usuario.

El gorro de electrodos, consiste en un casco de 32 electrodos los cuales miden las señales emitidas por el usuario para más tarde analizarlo, estos electrodos llevan un amplificador para poder ir viendo a tiempo real los valores de impedancia a tiempo real para poder así poder obtener las mejores señales, las cuales estarían por debajo de valores de 20 Kohm, para poder bajar estas señales se utiliza un gel electroconductor aplicado a cada electrodo, este gel baja reduce la impedancia con el pelo.

Se emplea una frecuencia de muestreo de 500 Hz para recoger todas las muestras emitidas por el usuario. Esta es una de las grandes diferencias y

cambios que se han introducido en esta nueva investigación, cuando se cambia esta frecuencia de muestreo se recogen los valores cada menos segundos. Estas señales de muestreo son recogidas mediante Matlab.



Figura 7. Casco electrodos.

El amplificador actichamp, el cual posee 32 canales, se usa para conectar las señales del gorro de electrodos al ordenador mediante una serie de cables, uno conector a tierra y el otro al ordenador, y mediante el programa Pycorder podremos ir analizando las señales de las pruebas para que no existan problemas y también poder ver, como he comentado antes, los niveles de impedancia de los diferentes electrodos del gorro.

Todos los registros obtenidos mediante los materiales previamente citados y los que se citarán posteriormente se han hecho a través del software de Matlab, en el que mediante una serie de scripts y diferentes programas se sincronizan todos los materiales y se consiguen los datos que queremos estudiar



Figura 8. Actichamp.

3.1.2 IMUs

Otro componente principal para estudiar los datos son las IMUs, consisten en sensores inerciales de detección de la orientación del cuerpo en base a las fuerzas que este recibe. Esto nos sirve para poder detectar exactamente cuando el usuario se detiene ante el objeto, así registramos los datos emitidos por el usuario ante objeto. Estas IMUs las tenemos que ajustar a una determinada frecuencia que será de 30 Hz.

Los sensores inerciales que componen estas IMUs son un acelerómetro, un giroscopio, un magnetómetro y un sensor de temperatura todos ellos controlados por un microprocesador.

Las señales que nos proporcionan estos componentes se fusionan para obtener la orientación de cada una de las IMUs.



Figura 9. Equipamiento IMUs.

Las IMUs usadas para este proyecto son las Tech-Mech V3, este equipamiento está constituido por los Tech-MCS studio, los Tech-HUB y las Tech-IMUs, además de los agarres para poder situar las Tech-Imus agarradas en las zonas que las queremos colocar para poder así evitar oscilaciones las cuales nos generaría pérdida de datos o simplemente ruido en nuestra señal lo que provocaría que no fuera válida.

El Tech-Hub es el dispositivo que se encarga de recibir la información capturada por las Tech-IMUs. Este dispositivo se forma por 3 diferentes modos, los cuales son un puerto USB, una conexión bluetooth y una ranura para un

almacenamiento mediante tarjeta SD. Al contar con cuatro puertos, podemos conectar hasta 16 Tech-IMUs a la vez.

Las Tech-IMUs son los sensores inerciales utilizados para la detección del movimiento. Son pequeños dispositivos que contienen dentro de sí un acelerómetro, un giroscopio, un magnetómetro y un sensor de temperatura.

El funcionamiento de estos componentes está basado en las dos primeras leyes de Newton, la primera ley nos ayuda a obtener el cambio de una persona del reposo al moverse, y con la ayuda de la segunda ley podemos obtener las variaciones de velocidad, aceleración, que tiene el usuario durante los diferentes momentos de las pruebas.

En nuestro proyecto haremos uso de 7 diferentes IMUs, las cuales estarán situadas 2 en la zona del muslo de cada pierna, otras dos a la altura de la rodilla de la persona, otras dos más en la zona de los tobillos y para finalizar una en la zona lumbar, en la figura 10 se puede apreciar la colocación de estas.

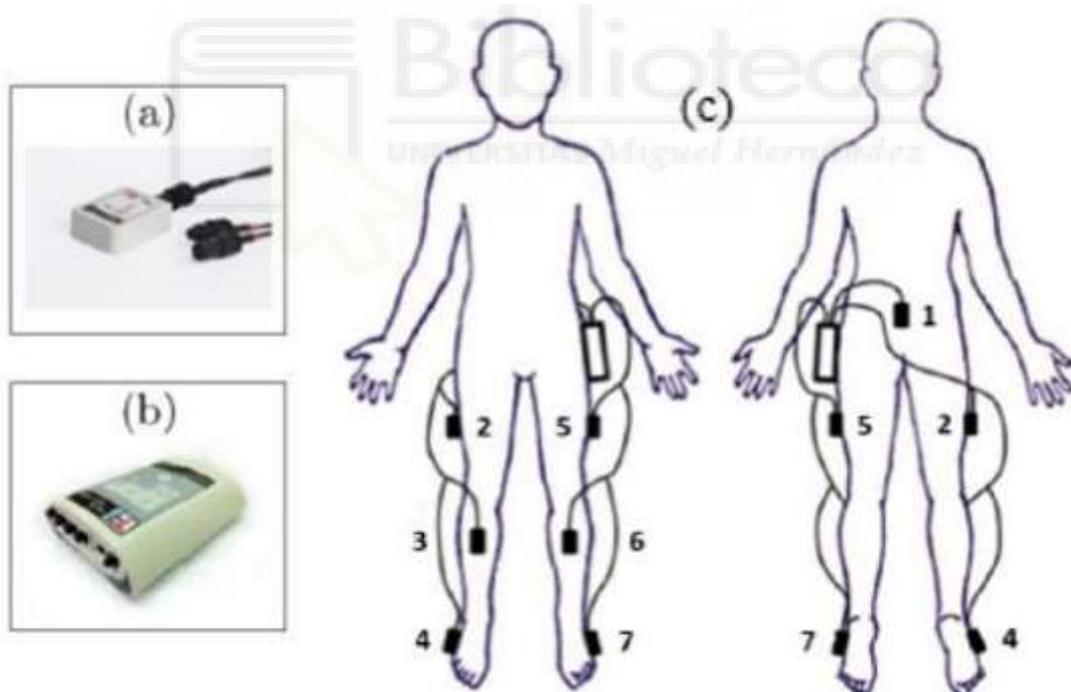


Figura 10. Posición IMUs [13].

3.1.3 Láser y cinta de correr.

En nuestras pruebas, aparte de los elementos citados previamente, hemos hecho uso de dos elementos indispensables para poder realizar las pruebas y poder así obtener los datos como son el láser y la cinta de correr.

La cinta de correr, de la marca Performance 750, nos ayuda a poder simular el recorrido y así poder caminar sin ningún obstáculo y sin ninguna parada para así obtener unos datos lo más limpios posibles. Para que el usuario pueda caminar tranquilamente y sin ningún se establece la velocidad de esta a 2 km/h, a esta velocidad se le permite al usuario poder detenerse sin riesgo a caerse por la parte trasera de la cinta. La cinta tiene dos modos disponibles para conectarla con el ordenador, una mediante Bluetooth, que es el que usamos en nuestras pruebas, y un segundo modo manual. Esta selección se hace mediante un Arduino.

El láser, el cual este compuesto por una línea láser y un arduino que nos permite conectarlo con el ordenador para poder así usarlo. Este sistema de electrónica está diseñado por el grupo Brain-Machine Interface System Lab y se ha utilizado en diversos proyectos anteriores. El láser se enfoca en la parte delantera inferior de la cinta de correr sobre una cinta adhesiva blanca, para que así se pueda reflejar mejor y el usuario pueda verlo sin problemas.

Se usa para poder así simular un obstáculo para que el usuario se detenga y así poder analizar y obtener los datos para más tarde estudiarlos. Este láser está controlado mediante una función de Matlab que he diseñado para poder emitir una señal aleatoria que más adelante explicaré.



Figura 11. Montaje experimental.

3.2 Protocolo experimental para registro.

Para realizar nuestra investigación se requiere implementar un protocolo a la hora de actuar con el usuario para poder tener todo clase y atado a la hora de las pruebas.

3.2.1 Protocolo experimental offline.

Para poder realizar este proceso experimental, al usuario que se presta voluntario para realizarlo se le explica detalladamente la prueba, una vez realizado este paso se le da una hoja para que firme su consentimiento. Este consentimiento consiste en una hoja en la cual nos permite realizar la prueba y utilizar la información obtenida bajo su consentimiento y además existe una cláusula la cual permite al usuario parar la prueba cuando el desee y finalizarla.

En el proceso experimental, primeramente, se le mide el perímetro craneal al sujeto y se le ajusta el gorro a su medida. A continuación, se conecta dicho gorro al ordenador para poder ir obteniendo los datos e ir ajusto los datos de las impedancias en tiempo real. Para finalizar esta primera parte, se le inyecta a cada electrodo el gel electroconductor para reducir los niveles de impedancia y obtener los datos más precisos, los niveles de impedancia de cada electrodo deben estar por debajo de los 20KOhm, como citamos en el punto 3.1.1, hay que aplicar un gel para reducir la impedancia con el pelo, este proceso tarda en torno a los 45 minutos, ya que depende del tipo de cuero cabelludo de cada persona o del estado del pelo o forma de la cabeza, que el proceso se demore más tiempo o menos.

Una vez realizado este proceso, se comienza a colocarle al usuario las IMUs en las posiciones que se muestran en la figura 11, y una vez finalizado, el usuario se coloca en la cinta para comenzar con la prueba.

Ya con el usuario en la cinta, se conectan tanto la cinta de correr como el láser. Cuando ya está todo conectado y operativo, el usuario se queda parado y quieto en la cinta, para que así las IMUs se calibren y no nos ofrezcan datos falsos o no válidos.

A continuación, se lanza el programa y la cinta empieza con la primera tarea, la cual es la tarea 400 que son 15 segundos a 0 km/h para que así el programa empiece a calibrarse y se ejecuten los primeros algoritmos, para seguir con la segunda tarea que es la 500, la cual es el inicio de la cinta a una velocidad reducida para así evitar un inicio brusco, esta tarea dura 15 segundos, durante el proceso de esta tarea el usuario comienza a caminar y establece la velocidad promedio para luego la siguiente tarea. La tarea principal (600) viene a continuación, y consiste en establecer una velocidad constante de 2km/h durante 90 segundos en los que el láser se irá activando de forma aleatoria en un período de 6 a 9 segundos en los que el usuario se detendrá, para así poder evitar la monotonía de la señal y que el usuario se espere cuando sea el cambio, lo cual provocaría variaciones en las decisiones y así en los datos.

Tras cada emisión de láser el usuario se detiene en la cinta y reanuda la marcha de nuevo para continuar con la tarea y así poder volver a detenerse con el nuevo láser que se emita. El periodo de detección del usuario es muy corto, ya que se realiza con la cinta de correr en marcha y si fuera más largo de lo normal este se podría caer de la cinta.

Una vez finalizados los dos minutos de la prueba se recogen los datos dentro de una estructura, la cual registra toda la información tanto de las IMUs como de los electrodos, esta estructura luego se usará, junto con todas las que componen el conjunto de la prueba, para obtener los datos que estudiaremos.

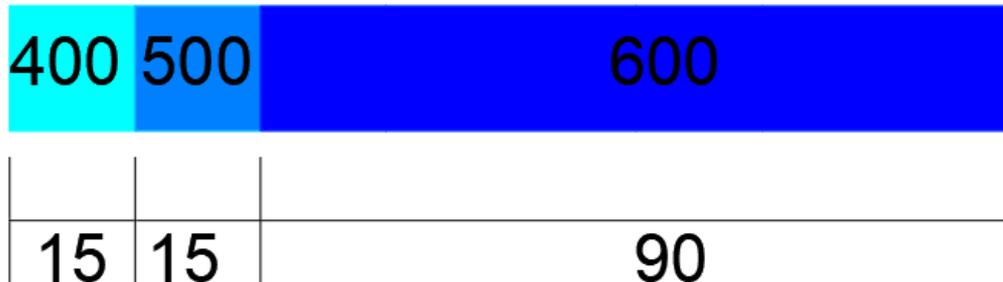


Figura 12: Tareas (400, 500, 600) y tiempos en segundos del modo offline.

La prueba se ejecutará entre unas 10 y 14 veces para poder así eliminar las pruebas que nos supongan erróneas, estas pruebas pueden ser las cuales al usuario se le cae o resbala de la pierna una, o realiza un movimiento no convencional con el cuerpo el cual generaría ruido o mala recepción de nuestra señal, por lo que así podemos elegir de las sesiones más depuradas, con el fin de obtener unos datos mejores.

Cada prueba realizada contiene aproximadamente 10 láseres emitidos, por lo que, si hacemos unas 10 pruebas, tendremos 100 eventos de láseres distintos para poder analizar y posteriormente estudiar.



Figura 13: Usuario W30 durante la realización de una prueba

3.2.2 Función laser_vector

Una de las grandes aportaciones que he realizado a esta arquitectura y por tanto al proyecto es la creación de la función laser_vector que nos ayudará en nuestras pruebas offline a obtener resultados.

La principal función de este script es la generación de un láser que se emitirá en un período de tiempo aleatorio entre 6 y 9 segundos, esto quiere decir que la primera generación puede realizarse a los 30 segundos de la prueba y la siguiente aparición del láser dentro de la prueba puede producirse en una ventana de entre 36 y 39 segundos no siguiendo un patrón. También se configuró la señal del láser para que esta durara un tiempo (time_on) de x segundos que nosotros le indiquemos para que al usuario le dé tiempo a detectarlo, frenarse y volver a retomar el ritmo.

Esta función se encuentra dentro del ConfAcquisition, además desde ese mismo script se le pueden ir variando los valores de la función, tanto la ventana de tiempo que queremos que se generen los láseres como el tiempo que queremos que duren a nuestras necesidades a cubrir.

En la siguiente imagen se puede observar gráficamente el funcionamiento de la función para una frecuencia (Fm) de 500, un shift de 250, lo cual usamos como desplazamiento y es cada cuanto avanza el usuario, y un time_on de 2 segundos, además se observarán los 30 primeros segundos que son los de calibración de artefactos más inicio de la secuencia para acostumbrar al arranque al usuario y los 90 segundos siguientes de la prueba con las 10 emisiones de los láseres.

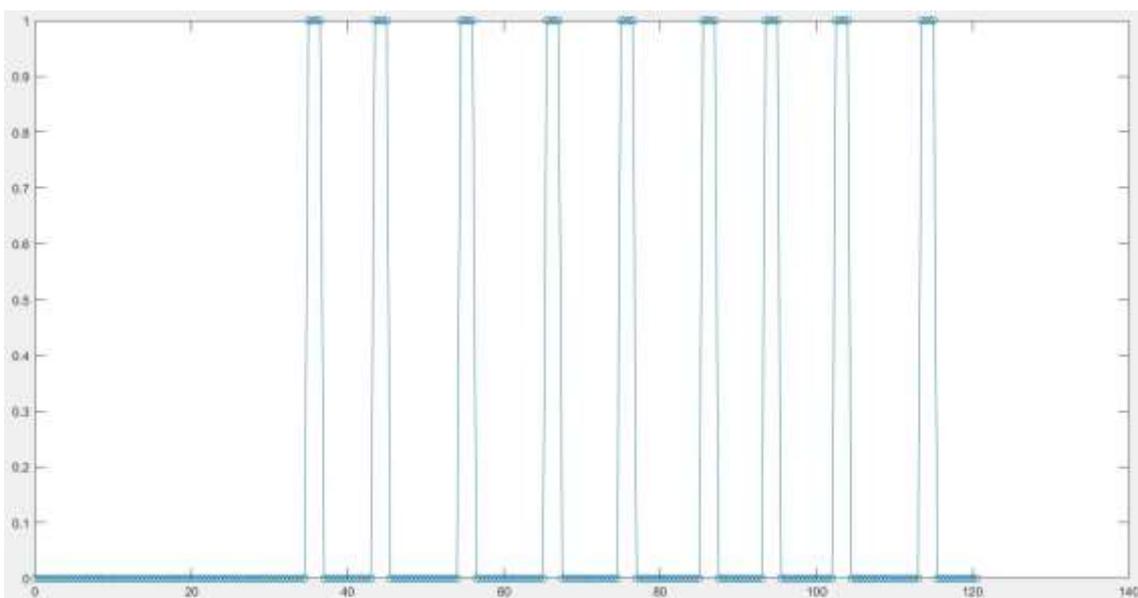


Figura 14: Resultado función laser_vector

3.3 Método automático para la detección de la parada mediante IMUs.

Para saber cuándo se ha parado el usuario y para conocer el rango en el que debemos analizar las señales ha sido necesario detectar el momento en el que el usuario realizaba la parada, con el resultado de poder asegurar la respuesta del sistema BMI fuese anterior a la reacción normal de las personas.

Para ello ha sido necesario el uso de las 7 IMUs, distribuidas y usadas como comentamos en el punto anterior y obtener el valor del tiempo medio de parada que se aplica antes de llegar a la parte del pseudo-online.

Para realizar este proceso y poder conocer todas las paradas generadas por las IMUs usaremos la función `findchangepts`, esta función ha sido desarrollada por un compañero de investigación en un TFG previo [10]. Esta función se ha ido desarrollando durante varios TFGs anteriores para finalmente usarla en este [11].

Esta función consiste en que nos devuelve el índice en el que la media de los cambios producidos es más significativa, te devuelve los puntos donde se producen cambios bruscos en la señal. Esta herramienta nos es muy útil a la hora de detectar de forma automática el cambio en la parada.

Anteriormente disponíamos del método CWT, el cual consistía en un análisis en tiempo-frecuencia simultáneamente, la mejora respecto a este método recae en la cantidad de detecciones que tenemos, ya que antes no podíamos medir todas las detecciones que queríamos.

Por lo tanto, con este nuevo método solo se le pasan ventanas en las que sabemos que se encuentra la parada y ya el solo se encarga de detectar el instante en el que se realiza dicha parada.

Como se puede observar en la siguiente gráfica se diferencia cuando el usuario está caminando, y cuando detecta la aparición del láser se realiza un cambio brusco y luego se genera otro cambio brusco cuando vuelve a reanudar la marcha.

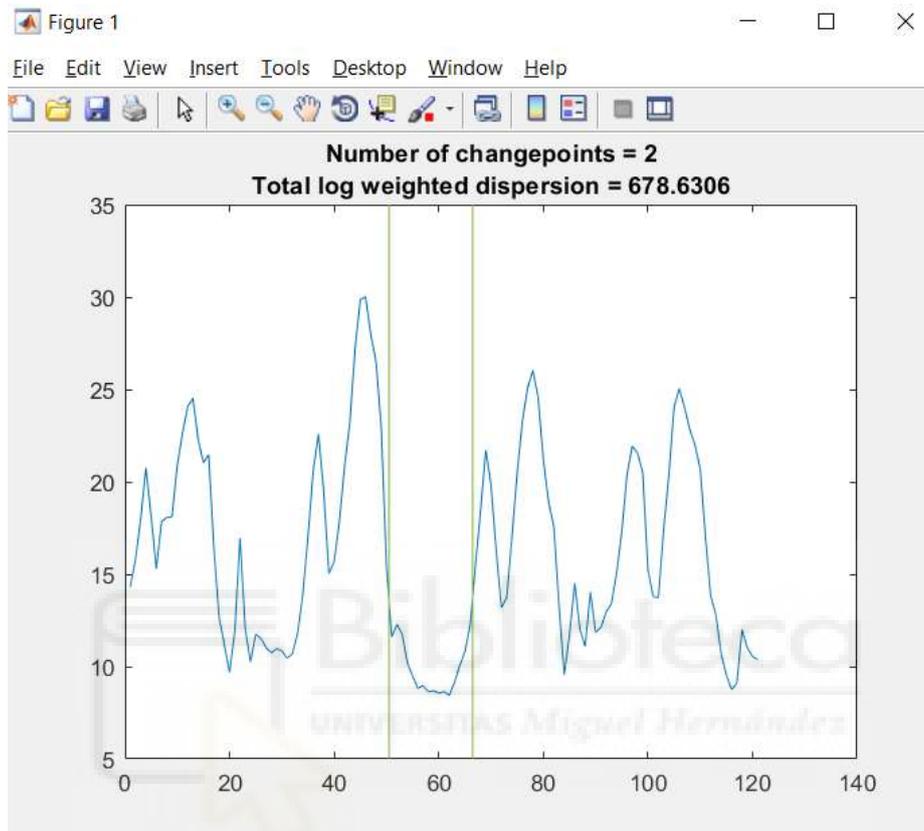


Figura 15. Resultado findchangepts correcto.

No todo es perfecto con el uso de este método, ya que se puede detectar un láser en el que no se detecte una parada, en este caso se nos devolverá un vector vacío. Otro problema generado con el uso de esta función es que para algunos láseres la detección no se produce en el instante que se produce la parada.

Tampoco es un método que se podrá usar en nuestras pruebas en tiempo real, por eso tendremos que general el modelo previamente con los usuarios que participen en estas pruebas, y luego cargarlo y aplicarle el código que utiliza el findchangepts para poder así usarlo sin problema.

Para solucionar este problema, se ha realizado el cálculo de la mediana de todas las paradas del usuario y sustituirlo por esta, así todos los los puntos conflictivos

que nos encontramos durante diferentes tramos de la prueba tengan una variación elevada en comparación con la mediana calculada para el resto.

Así es como se vería un fallo en el findchangepts.

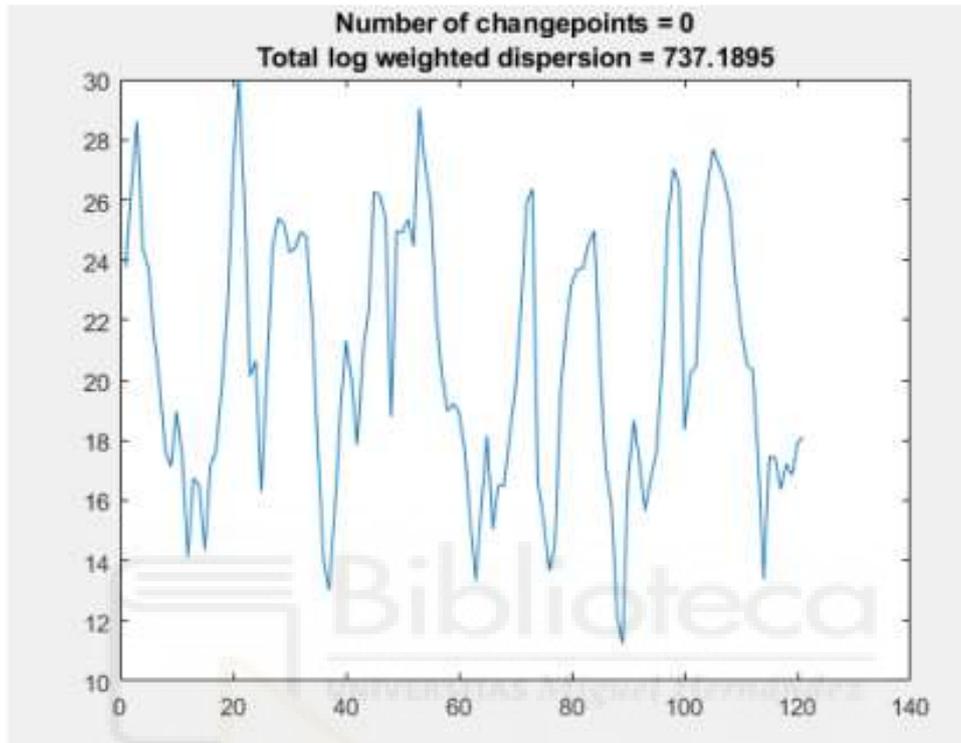


Figura 16: Resultado findchangepts incorrecto

3.4 Análisis de la actividad cerebral.

A continuación, describimos las distintas etapas en las que consiste nuestros procesamientos de las señales que vamos obteniendo de los usuarios, así como la estrategia a seguir en las distintas etapas.

Estas etapas se basan en los procesos realizados por mis compañeros anteriores [10], que hemos utilizado y añadido algunas mejoras que posteriormente comentaremos. [11] Se comienzan por la carga de sesiones, para continuar con el preprocesamiento, la selección automática, seguir con la selección de clases y finaliza con la aplicación del pseudo-online.

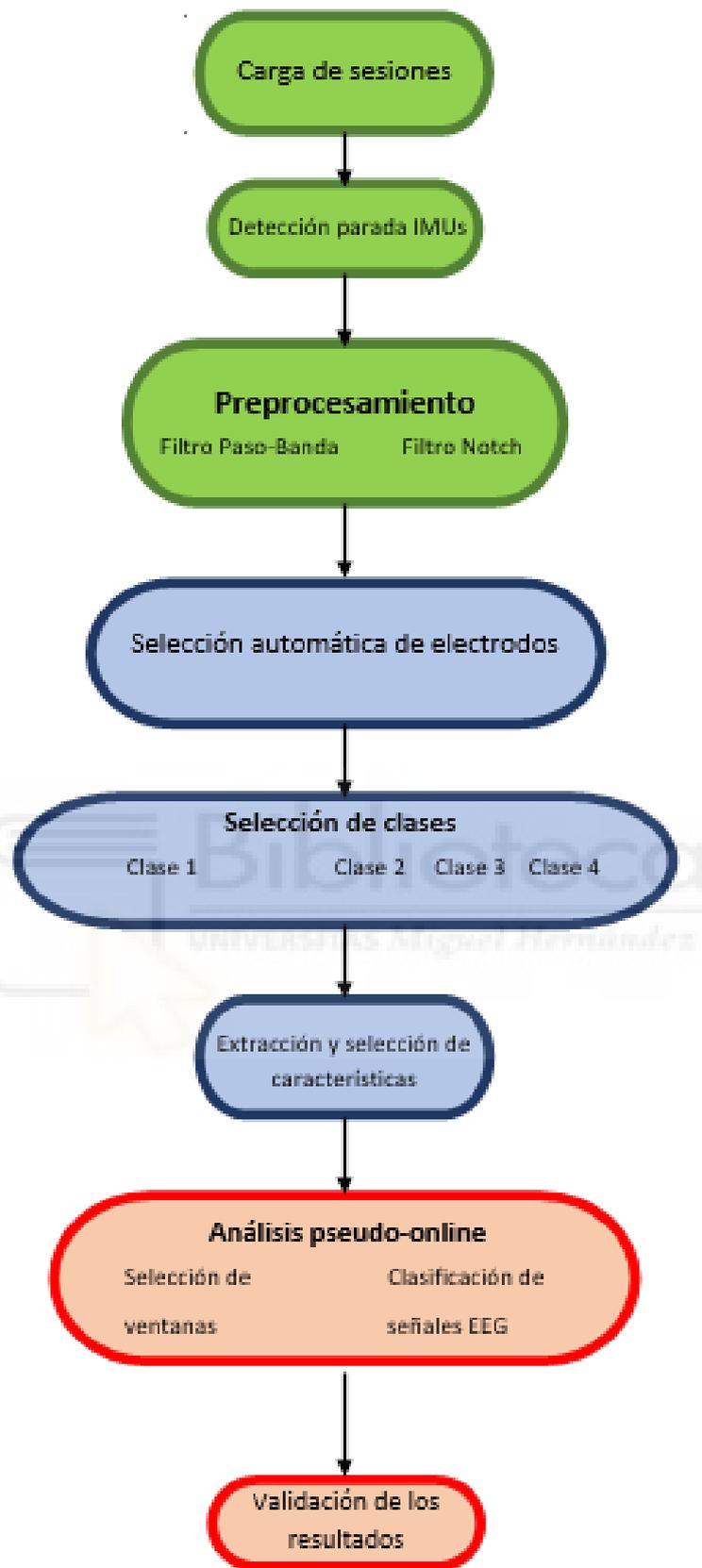


Figura 17: Esquema del modelo empleado.

3.4.1 Preprocesamiento

Para realizar el preprocesamiento de nuestras señales EEG, primeramente, se le aplicará un filtro Notch, el cual estará fijado a una frecuencia de 50 Hz, que se usará para la eliminación del ruido en la señal EEG.

Posteriormente, se analizan los electrodos que usaremos y los electrodos que eliminaremos los cuales se encuentran en la periferia de la cabeza y pueden verse afectados por el aumento de ruido en la señal en dichas zonas. Los electrodos eliminados son los mostrados en el mapa de electrodos siguiente, coloreados de color rojo.

Para la realización de las pruebas, como hemos hablado en puntos anteriores, usaremos un casco de electrodos con 32 electrodos posibles, de los cuales eliminamos 10 electrodos, por lo que nos quedaremos con 22 electrodos, que se escogerán de manera automática mediante la selección automática de electrodos dentro del propio código.

Se aplicará un filtro paso-banda de 0.4 a 3 Hz [4], ya que las señales EEG por la aparición de obstáculos se manifiestan a frecuencias bajas, por eso se escoge un filtro con una frecuencia tan baja.

Por último, se aplicará un filtro CAR para suavizar las contribuciones de la actividad cerebral basal [13].

3.4.2 Selección de electrodos

Nuestro cerebro está dividido por lóbulos, los cuales realizan una función concreta, las divisiones de los lóbulos se denominan frontal, temporal, parietal y occipital.

Con esto podemos, para nuestras pruebas, elegir y explotar la zona que más nos convenga obtener señales y por se obtener información y estudiarla del lóbulo que queramos. Para nuestro proyecto, la zona principal que queremos explotar serán las zonas motoras, sensitiva y occipital, localizadas en el lóbulo occipital, frontal y parietal.

Para poder conseguir esta selección a explotar, usaremos un método de detección automática de electrodos, el cual nos ayudará a detectar los electrodos que tienen mayor diferencia entre el instante anterior al laser y el instante posterior a este.

Los electrodos escogidos para las diferentes pruebas son los siguientes:

Fz, F3, FC5, FC3, FC1, FCz, FC2, FC4, FC6, C3, C1, Cz, C2, C4, CP3, CP1, CPz, CP2, CP4, P3, P1, Pz, P2, P4, PO3, Poz, PO4, OZ.

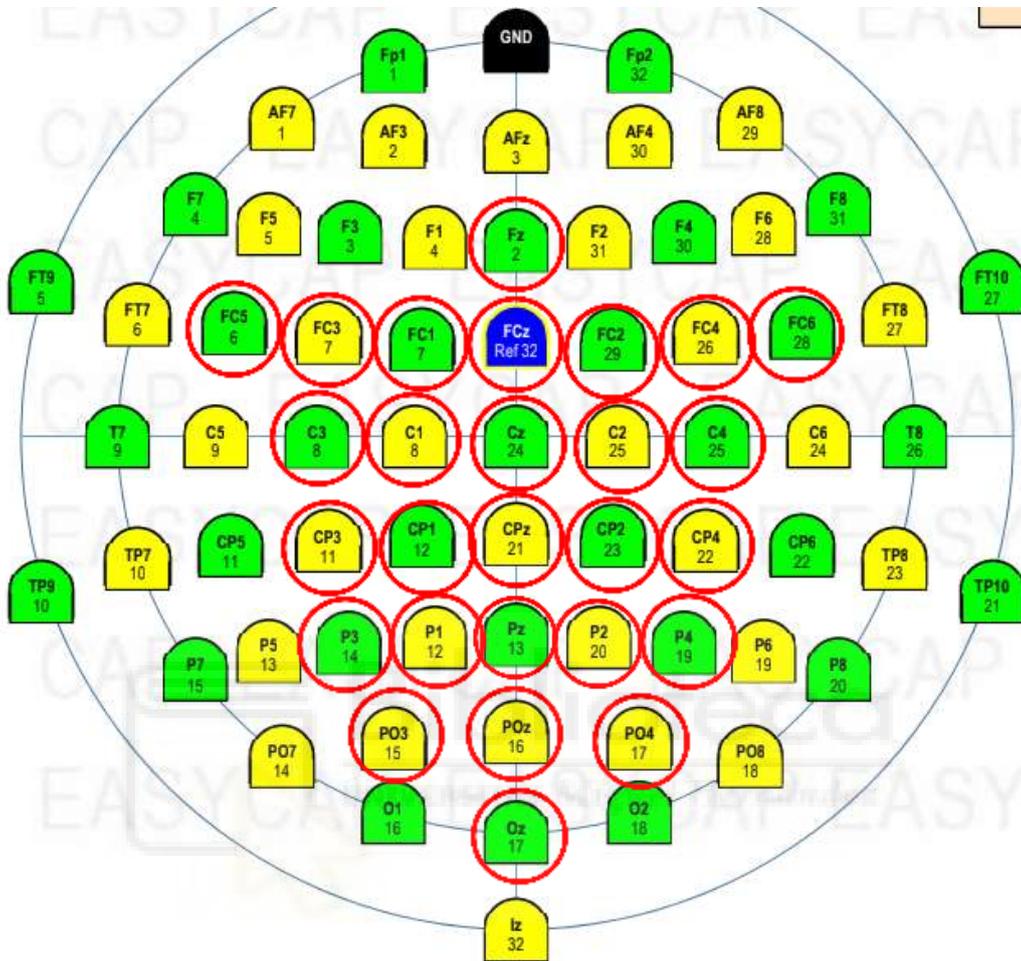


Figura 18: Mapa electrodos con la selección

Además de todos estos electrodos, usaremos 4 electrodos más los cuales situaremos en la cara. Situaremos uno encima del ojo izquierdo, uno encima del ojo derecho y otros dos en las partes laterales de ambos ojos. El objetivo de la colocación de estos 4 electrodos es la eliminación de la contaminación que puede producir el movimiento de ojos y cara durante las pruebas, así podemos monitorizar dicho movimiento para después eliminarlo y obtener una señal más limpia.

3.4.3 Selección de ventanas

Los ERP presentan una gran variabilidad entre los distintos individuos, esto nos quiere decir que los tiempos de reacción que presentan los usuarios frente a los obstáculos son diferentes. Esto nos genera un problema a la hora de poder escoger las ventanas posibles para la obtención de la detección de ERP [12].

Como se puede observar en las siguientes gráficas de 2 usuarios los tiempos de reacción frente a frente varían ampliamente, donde en el eje x se representa el tiempo de las muestras, y en el eje y podemos ver la variación del potencial. En el instante 0 que se representa se trata de la aparición del láser. Y con un distinto grosor se puede ver la media de los potenciales.

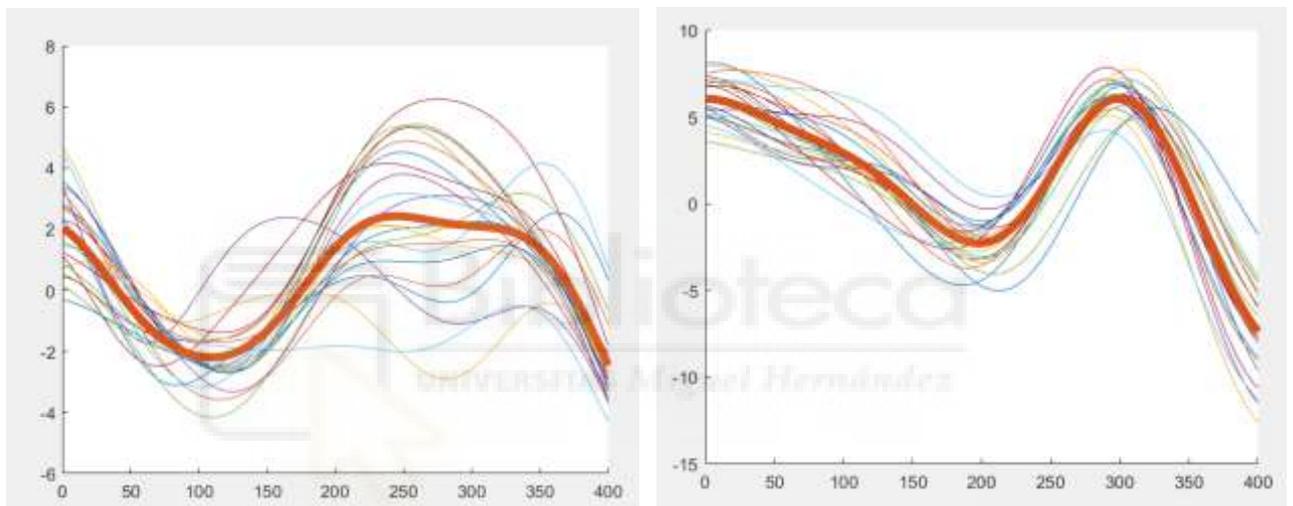


Figura 19: Comparación muestra usuarios W55 y W30

Por lo tanto, generaremos selección de ventanas distintas para las tareas que queramos realizar.

3.4.3.1 Selección ventanas para la clase 1

Para obtener la clase 1 en esta selección de ventana [4], se calcula a partir del tiempo medio de cada usuario, es decir en el cual se encuentra el pico de la señal tras la aparición del láser.

La selección de estas ventanas mejora en enorme medida la detección de las paradas, pero también los ERP presentan una cierta variabilidad. Por lo que se plantea como solución la posibilidad de sacar el tiempo para cada uno de los láseres de forma automática y personalizada.

El único inconveniente que se presenta es que no todos los láseres emitidos tienen el pico máximo en el lugar que debería, por lo que se calcula la mediana de todos los puntos máximos de cada señal de láser y ya así se obtiene el valor que deseamos.

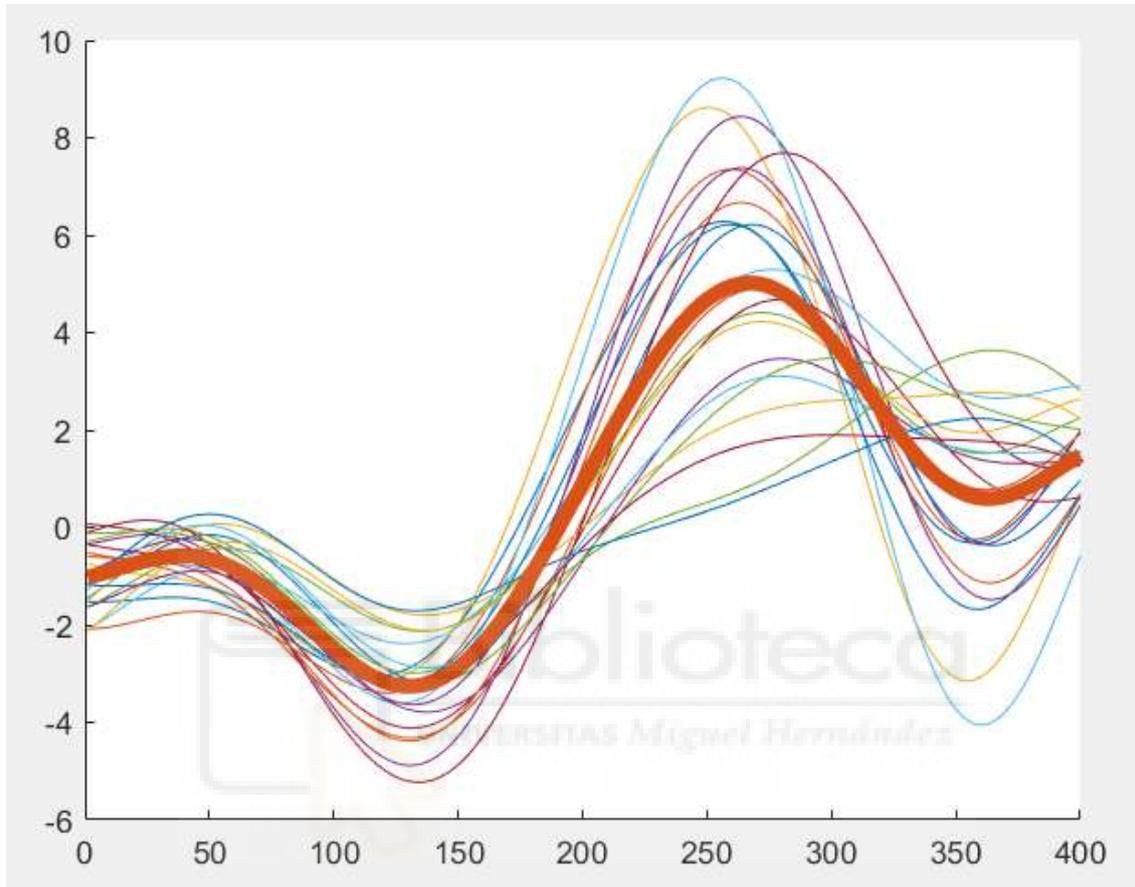


Figura 20: Representación clase 1 usuario W26

3.4.3.2 Selección ventanas para la clase 2

La clase 2 corresponde a la actividad de caminata normal, para esta clase usaremos diferentes datos de entrenamiento obtenidos durante la marcha normal.

Para la extracción de la clase, nos bastará con la extracción de las ventanas en los momentos en los que interpretemos que no están produciendo ningún ERP, por eso es el motivo por el cual se han escogido momentos en los que se evita la aparición del estímulo visual.

Esta clase se usa unos instantes previos a la aparición de láser y se cogen muestras las cuales tienen lugar dos segundos antes del láser, una ventana de 1000 muestras.

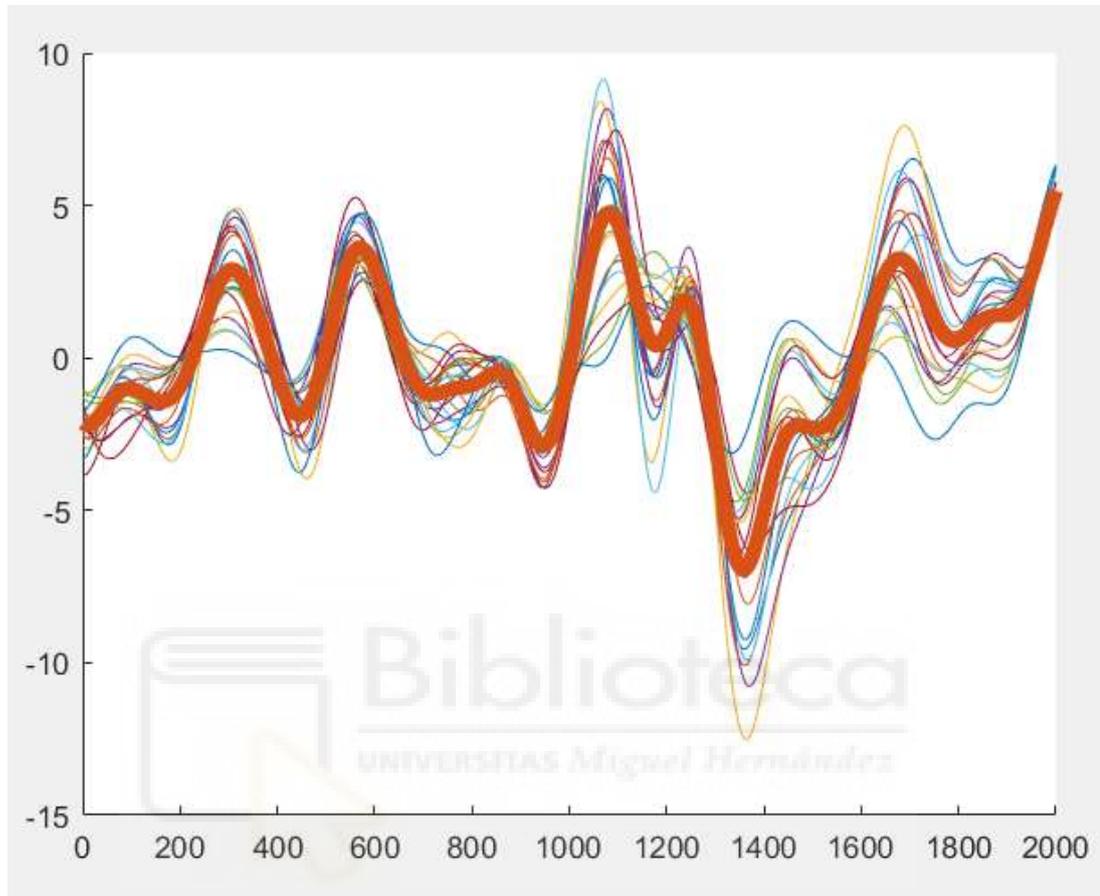


Figura 21: Representación clase 2 usuario W26

3.4.4 Extracción de características

El mayor problema que se nos presenta a la hora de diseñar un sistema BMI es la gran cantidad de datos existentes a la hora del inicio de la actividad, de los cuales muchos se podrían desechar por ser insignificantes, por ello resulta casi imprescindible escoger un subconjunto de características las cuales nos permitan poder distinguir entre las distintas clases de la señal para poder elegir las para elaborar un futuro modelo.

Estas características se pueden dividir en dos grupos distintos para poder así clasificarse, por una parte, tenemos las del dominio de la frecuencia, que se centran en los estados mentales del sujeto, y las del dominio temporal, que serán con las que nosotros trabajaremos.

Para nuestro estudio, al tratarse de un ERP (potencial relacionado con un evento), 2.2.3, caracterizado por una variación en la amplitud de la señal EEG, ligado a la aparición de un estímulo, por lo que los métodos de extracción de las características están basados en el dominio temporal. Como la finalidad de esta investigación es la aplicación de los métodos a tiempo real, este tipo de método es menos costoso en cuanto a tiempo que el cálculo de una FFT o CWT.

En la investigación se han ido probando con diferentes números de características, desde 17 hasta 21, para finalmente quedarnos con las 21 características, ya que no todas las características son igual de buenas a la hora de detectar una parada.

Cada característica posee unas condiciones especiales y nos aportan unos determinados datos a la hora de sacar nuestros resultados, en la siguiente tabla veremos el total de las 21 características con sus nombres y su función correspondiente en Matlab, ya que es ahí donde las calculamos.



Tabla 1: Número de características

CARACTERÍSTICAS

1. ERROR CUADRÁTICO MEDIO	<code>RMS1=rms (suma_laser)</code>
2. DESVIACIÓN ABSOLUTA MEDIA	<code>MNAD1=mad (suma_laser,0)</code>
3. MEDIANA DE LA DESVIACIÓN ABSOLUTA	<code>MDAD1=mad (suma_laser,1)</code>
4. FRECUENCIA MEDIA	<code>FMN1=meanfreq (suma_laser, fm)</code>
5. AMPLITUD DE LA SEÑAL	<code>AMP1= peak2peak (suma_laser)</code>
6. VARIANZA	<code>VAR11= var (suma_laser,1)</code>
7. DESVIACIÓN ESTANDAR	<code>STD11= std (suma_laser,1)</code>
8. ERROR SEÑAL MEDIA CWT	<code>DWT1= dtw (suma_laser, signal)</code>
9. INTEGRAL DEL LÁSER	<code>integral_laser= [trapz(1: length(derivada_laser),abs(derivada_laser))]</code>
10. VARIANZA DEL LÁSER	<code>varianza_der_laser=[var(derivada_laser)]</code>
11. AMPLITUD DEL LÁSER	<code>amplitud_laser=[max(derivada_laser)- min(derivada_laser)]</code>
12. COEFICIENTE DE CORRELACIÓN DEL LASER	<code>coef_corr_laser= [max(xcorr(sum(matriz_reaccion_media(ch1, :),1),suma_laser,long_ventana*fm,'coeff'))]</code>
13. ACUMULADA DEL LÁSER	<code>acum_laser= [max(suma_acumulada_laser)- min(suma_acumulada_laser)]</code>
14. AMPLITUD DE LA ENVOLVENTE	<code>ENV_AMP1= peak2peak(envelope(suma_laser))</code>
15. VARIANZA DE LA ENVOLVENTE	<code>ENV_VAR1= var(envelope(suma_laser))</code>
16. INTEGRAL DE LA ENVOLVENTE	<code>ENV_INT1= [trapz(1: length(diff(envelope(suma_laser))),abs(diff(envelope(suma_laser))))]</code>
17. DESVIACIÓN ESTÁNDAR ENVOLVENTE	<code>ENV_STD1= std(envelope(suma_laser),1)</code>

Además de las 17 características citadas, se añaden cuatro más que son la las generadas por la señal EEG tras la aparición del láser.

Cuando ejecutamos el código para los diferentes usuarios no siempre se cogen todas las características, ya que en TFG anteriores [10], se realizó un script para poder seleccionar las mejores características para cada usuario. Más adelante en el apartado resultados se verá para cada usuario cuales han sido las mejores características.

3.4.5 Clasificadores

El siguiente paso a las características es obtener los clasificadores, la eficacia de los clasificadores se encuentra en lo bien diferenciados que estén los datos gracias a lo realizado en el apartado anterior mediante las características. Disponemos de dos tipos de clasificadores distintos para poder operar, el clasificador de máquina de soporte de vectores (SVM) y el análisis discriminante lineal (LDA).

3.4.5.1 LDA

El sistema de clasificación LDA consiste en un método supervisado de variables cualitativas en el cual, los grupos son conocidos a priori y se van clasificando según las características que tengan, haciendo uso del teorema de Bayes, se estima la probabilidad y se clasificada en una clase u otra.

La principal diferencia que se encuentra usando este clasificador, es que se emplea 3 clasificadores en serie en lugar de uno como con otros clasificadores diferentes. Para los 3 clasificadores se usará la misma clase 1 que corresponde a la detección de parada, sin embargo, la segunda, correponderá a distintos momentos de la marcha.

En nuestro código usaremos el clasificador LDA, debido a que en los estudios anteriores [11] nos ofreció resultados ampliamente mejores que el clasificador SVM.

3.4.6 Análisis pseudo-online.

La finalidad de este proyecto de investigación es la aplicación de todo este sistema en tiempo real, en trabajos previos a este se realizaron análisis en modo offline, para poco a poco ir evolucionando a un sistema pseudo-online en cual se parece en mayor medida a un posible análisis en tiempo real.

Para la realización de este método, se empleará como entrada los registros realizados por los usuarios, en los cuales se encuentra toda la información registrada de la prueba.

De los registros que utilizamos para analizar se escogerán unos para entrenar (Trainfiles) y otros para realizar test (Testfiles), esta cantidad de archivos que queremos para cada uno de los tipos se lo asignaremos cuando introduzcamos la llamada dentro de Matlab.

El método consiste en seleccionar una ventana móvil de x longitud que se desplaza z muestras. De cada una que obtenemos se le extraen las características y se le aplica el clasificador para obtener el resultado de a que clase pertenece.

A continuación, se aplica una K que denominamos la moda, esta consiste en probar varias opciones de 1 a 5 para poder luego analizar en los resultados cual ha sido el más correcto.

3.4.7 Métricas

El resultado del análisis pseudo-online, nos da unos resultados como los de la tabla 2.

Siendo la K la moda empleada, los FP/min, los falsos positivos por minuto que es el tramo en el que se ha detectado una parada cuando no se ha producido dicha parada, una parte importante ya que suele ser una señal muy contaminada por el ruido que se le introduce, y el TP (%), el porcentaje total de láseres detectados entre los láseres totales.

Tabla 2: Explicación datos resultado código pseudo

K	FP/MIN	TP (%)
1	0.81	20
2	1	12
3	1	10
4	1	8
5	1	8

3.5 Utilización arquitectura

Para la realización de estas pruebas, se han usado los materiales que se han expuesto previamente en el apartado 3.1 y se ha usado el protocolo experimental offline explicado en el apartado 3.2.1.

La realización de estas pruebas requiere la utilización de una arquitectura muy compleja que se ha ido desarrollando y evolucionando en pasadas investigaciones por compañeros y que hemos ido mejorando en esta investigación.

Esta arquitectura la denominamos BMI_register, y está compuesta por diferentes funciones y subfunciones que lo componen, algunas de las más importantes son las siguientes.

La función principal es la que denominamos Main, es la que ejecutamos y va llamando al resto de funciones para que realicen su trabajo. Cuando la hemos ejecutado y nos salta algún problema o error tendremos que ejecutar la función disconnect_app, la cual es muy útil ya que nos desconecta todos los elementos los cuales la función main ha introducido dentro del código y lo deja limpio para poder ejecutarlo otra vez.

Dentro de la carpeta Conf, encontraremos la segunda función más importante del código como es ConfAdquisition, en esta función se le dan todos los valores importantes a los elementos que usamos para realizar nuestras pruebas como pueden ser el láser, la cinta de correr... Esta función nos permite poder realizar nuestras pruebas en un modo de simulación, en el cual podemos probar si funciona todo el código sin necesidad de tener un usuario para realizarlas. Dentro de esta misma carpeta encontramos el resto de Conf, como pueden ser el ConfClassification o el ConfProcessing.

Como hemos citado previamente, dentro del ConfAdquisition se le asignan los valores a todos los materiales que componen nuestras pruebas, pero deben existir scripts que controlen todos estos elementos, y estos se encuentran dentro de la carpeta Adquisition, dentro de esta carpeta podemos encontrar todos los códigos que componen los objetos que vamos a usar, como el láser o la máquina de correr.

Estos scripts comentados previamente pueden ser los más importantes dentro de nuestro código, además existen diversas funciones aparte de estas como pueden ser Classification o Preprocessing.

3.6 Mejoras realizadas en los códigos

Para realizar la mayor parte de las pruebas el código estaba ya realizado ya que es un proyecto de investigación y como he citado previamente en TFGs anteriores se ha ido desarrollando, pero yo le he introducido alguna mejora [10].

Una de las mejoras introducidas es un código el cual nos permite poder realizar pruebas tanto con los usuarios antiguos como con los nuevos. El problema residía que en el vocabulario de las variables había un gran cambio con respecto a los usuarios antiguos y los nuevos, por lo que a la hora de probar con usuarios nuevos o antiguos había que hacer una gran serie de cambios.

Todo esto ha sido reducido a un script el cual te realiza todos los cambios necesarios automáticamente sin tener la necesidad de modificar nada.

Además, como se ha creado una función para que nos pueda generar el modelo con todas las variables necesarias para las pruebas de modo online, a esta función se le introducen los datos necesarios de cada usuario y crea una estructura en la cual se almacenan los datos necesarios que requiere la arquitectura de modo online.

Además, se ha generalizado el código para que pueda trabajar con las dos frecuencias de muestreo que usamos dentro de esta investigación, 0.1 y 0.2. Este tiempo de muestreo se utiliza para poder registrar y realizar todo el análisis (clasificadores, características, etc.) en ese mínimo tiempo para no obtener desfases en los comandos, por ello lo hemos adaptado para poder probarlo con ambos valores.

Se han creado tres funciones las cuales agrupan los filtros, clasificadores y características, esto se hizo para poder introducirlas de una manera más sencilla en la arquitectura online.

3.7 Protocolo online

Una vez realizadas las pruebas offline, procedemos a realizar las pruebas online. En estas pruebas se usan los mismos materiales que se usaremos previamente en las pruebas offline, la diferencia entre ambas pruebas reside en el proceso de estas.

El protocolo de los experimentos online consiste en realizar primeramente una serie de registros de entrenamiento igual que realizamos en las pruebas offline, explicado en el punto 3.2, seguidamente guardaremos estos archivos y generaremos un modelo, mediante el código que explico en el punto 3.5, con

los códigos que disponemos para poder utilizarlo en la siguiente etapa de bucle cerrado.

Una vez generado el modelo, este se introduce dentro de la arquitectura de bucle cerrado, entonces se realizan las pruebas en bucle cerrado, donde le mandamos mediante Matlab y por teclado órdenes a la cinta para que arranque o siga parada.

Una vez se ha mandado la orden a la cinta para que comience a funcionar, la secuencia de la realización es la siguiente, primero habrá unos segundos, entre 5 y 8, para que el usuario se acostumbre al arranque de la cinta y establezca la marcha sin sorpresas. Después de dicha parte, habrá otra ventana de unos 18 segundos en la cual se emitirá un láser y veremos si lo podemos detectar antes de que el usuario se frene ante la aparición de este. Si lo detectamos antes que el usuario, se le mandará un comando a la cinta y está se frenará, en cambio si el usuario lo detecta antes que nosotros, contará como un FP y tendremos que comenzar con el proceso otra vez. Esta secuencia se repetirá un número determinado de veces para poder realizar el estudio y aplicar los códigos más tarde para estudiar los resultados.

Esto, al igual que en el protocolo anterior está dividido por una serie de tareas diferentes para poder asignarles tiempos y poder ir ajustando las cuando sea necesario. Las tareas que se han usado han sido 4 y tienen los nombres de 400, 500, 601 y 602.

Las tareas 400 y 500 cumplen la misma función que se realiza en el protocolo experimental offline, la tarea 400 tarda 15 segundos, y durante este tiempo se va convergiendo un código. Después la tarea 500, establece el puente entre la siguiente tarea.

Las 2 nuevas tareas que se han añadido son la 601 y 602, la tarea 601 es la tarea en la cual arrancamos la cinta y le hacemos al usuario comenzar a andar. En este protocolo nosotros, manualmente desde Matlab, arrancamos la cinta mediante un comando, con el previo aviso al usuario de que va a comenzar a moverse.

La tarea 602 es la tarea en la cual se pueden comenzar a generar los láseres, estos al igual que en el protocolo experimental offline se producen mediante una franja de tiempo aleatoria, por lo que así el usuario no puede predecir la aparición de estos datos. El tiempo de tarea no está establecido ya que depende de la detección del código de las intenciones mostradas por el usuario a detenerse o no, por lo que cada usuario puede tener una duración determinada para esta tarea.

En este protocolo, también mediremos los FP/min y el TP (%) que se producen por los usuarios. Hasta que se muestre el láser y el usuario lo detecte todo lo mandado como comando de parada a la cinta es incorrecto por lo tanto contaría

como FP/min, en cambio si este comando de parada de cinta se envía cuando el usuario detecte la cinta se consideraría como correcto.

Como podemos ver en la figura 20, se diferencian las cuatro tareas previamente comentadas, este proceso se repite 2 veces por cada repetición, es decir por cada prueba que hacíamos con un usuario se reproducían dos tareas 400, dos tareas 500, dos tareas 601 y dos tareas 602.



Figura 22: Tareas protocolo Online

El objetivo principal sería detectar el láser antes que la persona para poder detener la cinta, para poder así controlar cuando el usuario decide frenarse y poder aplicarlo en un futuro a terapias de rehabilitación.





4. Resultados

En este penúltimo apartado se expondrán los resultados de los métodos expuestos en el apartado anterior y con la ayuda de dichos materiales.

La sección 4 estará dividida mediante el proceso que se ha realizado para obtener los resultados, desde las primeras pruebas con sus correspondientes resultados, hasta la adaptación a tiempo real y sus resultados obtenidos viendo si existe una mejora, además de las explicaciones de las pruebas y unos posibles trabajos futuros para mejorar el código y la forma de usarse.

4.1 Usuarios.

Para la realización de las primeras pruebas offline se ha utilizado a 3 voluntarios, los cuales 3 han sido hombres con una edad comprendida entre 21 y 24 años con ninguna complejidad física ni mental. Estos 3 usuarios eran todos diestros y ninguno de ellos tenía gafas ni lentillas. Estas personas fueron debidamente informadas del proceso de las pruebas, además tuvieron que firmar el consentimiento de la oficina del investigador de la UMH.

Aparte de estos 3 usuarios se han realizado otras 3 pruebas a otros 3 voluntarios en modo online, estos 3 nuevos usuarios tenían la edad de 22 años y han sido debidamente informados de la prueba a realizar además de firmar el consentimiento de la oficina del investigador.

Además, para comprobar las modificaciones de los códigos que comento en el punto 3.5, se han tenido que usar los usuarios que en TFGs anteriores realizaron las pruebas, que fueron 10 usuarios debidamente informados de las pruebas que se iban a llevar a cabo, la edad mediad de los usuarios que realizaron dichas pruebas se situaba en los 23 años. Estos usuarios también firmaron el consentimiento y no tenían ninguna complejidad mental o física [10].

4.2 Valores iniciales de dif_media y muestra

Para realizar estas pruebas aparte de lo comentado anteriormente, se aplica el protocolo experimental que comentamos en el punto 3.2.1, para poder ejecutar este protocolo necesitaremos de 2 valores que tenemos que extraer antes de poder ejecutarlo.

El primero de los valores es la dif_media, este valor corresponde al tiempo de detección de las paradas de las IMUs, como es previsible cada usuario tendrá un valor. Este valor lo obtenemos mediante un script desarrollado en TFGs

anteriores por mis compañeros, lo que realiza esta función es promediar todos los registros que le mandamos y hallar la media para obtener un valor. Este valor será dividido entre 30, el cual es el valor de frecuencia que tienen las IMUs, por lo que el resultado de esta división es el valor con el que nos quedamos.

Estos valores que nos generan pueden ser estables o no, debido a que, si son muy parecidos o poco, por lo que en la siguiente tabla veremos los valores para cada usuario y comprobaremos si son estables.

Tabla 3: Promedios dif_media de los 6 usuarios.

CÓDIGO USUARIO	DIF_MEDIA TOTAL	DIF_MEDIA PROMEDIADA	DIF_MEDIA EN SEGUNDOS
W26	[42.62 43.93 44.54 45.42 43.28]	43.92	1.46
W30	[42.18 47.06 50.74 52.09 54.71]	49.96	1.67
W33	[45.15 45.32 40.91 43.95 46.19]	44.44	1.48
W55	[48.94 50.61 45.37 46.71 49.11]	48.15	1.60
W56	[45.35 44.24 40.58 40.78 40.87]	42.36	1.41
W58	[44.50 46.06 46.57 46.90 46.05]	46.02	1.53

Como podemos observar en la tabla, los resultados son muy estables debido a que no varía casi a la hora de hacer la media y los valores son muy similares entre sí.

El segundo valor que necesitamos obtener es el valor de la muestra, este valor se obtiene de la clase 1 que hemos explicado en puntos anteriores. Este valor es importante para poder generar el modelo. Dentro de la obtención de este valor se encuentra una de las diferencias con las investigaciones anteriores, la diferencia reside en que la frecuencia utilizada en pasadas investigaciones era de 1200, y en nuestra investigación la frecuencia que escogemos es de 500. Esta frecuencia hace referencia a el sistema EEG. Esto nos aportará una muestra más pequeña.

Para la obtención de los valores, se realiza primero la obtención de los mejores electrodos de forma automática, esto nos permitirá seleccionar los electrodos que mejor información nos aporte para luego así poder generar la gráfica que nos da la muestra con la mayor exactitud posible.

En la siguiente imagen se puede observar la comparación entre obtener la muestra con todos los electrodos seleccionados y la obtención de la muestra usando únicamente los electrodos seleccionados por el método de obtención automática.

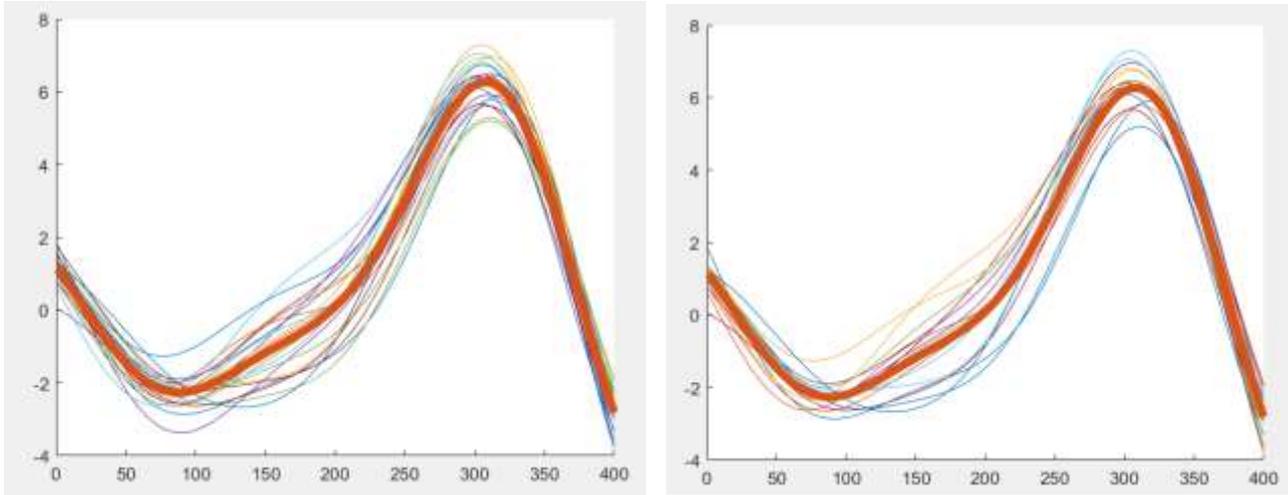


Figura 23: W33 obtención muestra código nuevo.

Como se puede observar en las gráficas, los valores que se obtienen para este usuario en concreto son muy similares por lo que no habría mucha variación en usar todos los electrodos o la selección a la hora de calcular la muestra.

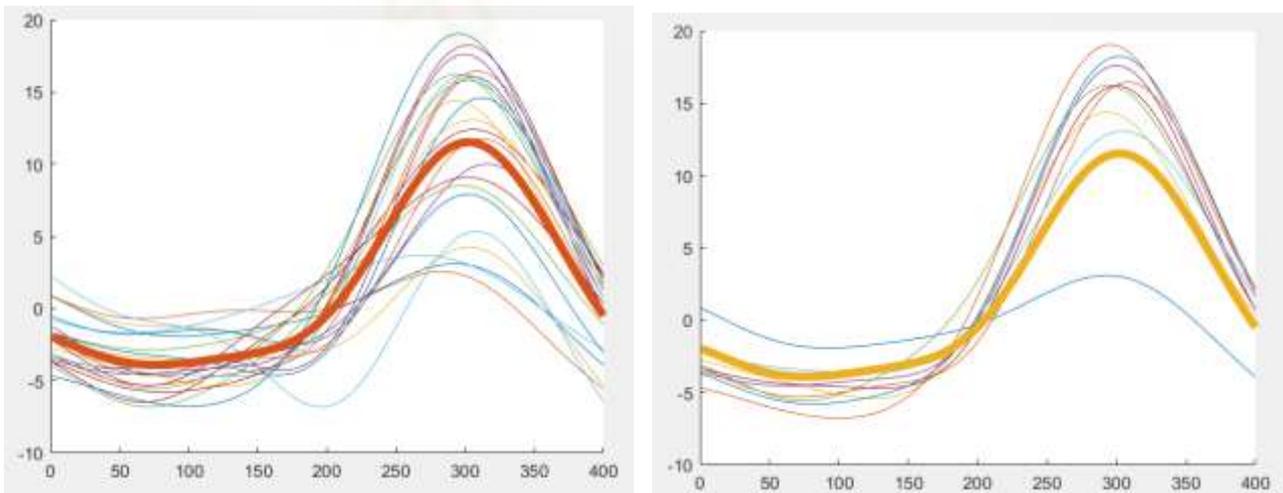


Figura 24: W58 obtención muestra código nuevo.

Aquí se puede diferenciar más la simplificación a la hora de la selección de los electrodos, pero el resultado final es muy parecido al que podemos obtener si usamos todos los electrodos.

Los resultados para los usuarios de las pruebas offline de los valores de dif_media y muestra son los siguientes:

Tabla 4: Resultados muestra y dif_media usuarios nuevos

CÓDIGO USUARIO	MUESTRA	MUESTRA (s)
W26	300	0.6
W30	265	0.53
W33	310	0.62
W55	240	0.48
W56	300	0.6
W58	320	0.64

4.3 Comprobación y adaptación código antiguo.

Una vez finalizadas las primeras pruebas con los usuarios que se comentan en apartados anteriores, comienza el estudio y la obtención de los resultados de estos usuarios, para ello tendremos que adaptar el código a las nuevas características y a la nueva configuración de los usuarios.

Durante la adaptación del código han ido surgiendo diversos problemas, ya que el nombramiento de los apartados dentro de las sesiones de los usuarios utilizados en proyectos anteriores es completamente distinto a la nueva configuración de los usuarios de las nuevas pruebas.

Dentro de los nuevos códigos usaremos la misma configuración que para código antiguo, esta configuración consiste en 6 diferentes parámetros, los cuales son, primeramente, la selección de los clasificadores en serie, que con el objetivo de reducir los FP/min se comprueba si introduciendo un mayor número de clasificadores nos permite hacer dicha reducción, en nuestro caso usaremos 3 clasificadores en serie.

A continuación, la siguiente configuración es la probabilidad a priori, está consiste en introducir al clasificador una probabilidad a priori para que, a la hora de clasificar, modifique los pesos que le da a cada clase, nosotros los usaremos, al igual que en la configuración uno, para que nos reduzca los FP/min, dado que en el pseudo-online, la cantidad de muestras empleadas para la marcha normal es mucho mayor que las ventanas que corresponden a la parada.

La tercera configuración que vamos a usar es el número de características que le aplicamos al código, debemos buscar cual es el número de características

óptimo que maximice los resultados para cada usuario, por lo tanto, nosotros usaremos el número de 21 características en esta configuración.

La siguiente configuración consiste en la selección de la muestra, la muestra hace referencia al número de intervalos de muestras que se sitúan entre la aparición del láser y el comienzo de la clase uno, en esta parte solo tenemos dos opciones de selección una selección que nos coja la muestra automáticamente y otra opción que la tengamos que introducir manualmente, tras estudiarlo llegamos a la conclusión de que usaremos una configuración automática.

La penúltima característica con la que nos encontramos a utilizar es la selección de electrodos, como comentamos en apartados anteriores, nos quedamos con los 22 mejores electrodos, en esta configuración podremos elegir entre una selección automática de los mejores electrodos que nos dan datos dentro de los 22 seleccionados previamente o poder realizar una selección automática de estos. Con un amplio margen, es bastante mejor utilizar una selección automática, porque sobretodo notamos una gran mejora a la hora de los TP (%).

Gracias a la mejora que se ha realizado para poder comprobar los resultados entre usuarios antiguos y usuarios nuevos, podemos realizar una comparación entre los resultados de la selección de electrodos entre los usuarios antiguos y los usuarios nuevos.

Tabla 5: Electrodos por usuario nuevos

SELECCIÓN ELECTRODOS	
W26	[1 2 3 4 5 6 8 9 11 12 13 18]
W30	[1 2 4 11 15]
W33	[1 5 6 8 9 10 11 12 13 15 17 18 20 21 22]
W55	[1 2 5 6 13 16 18]
W56	[1 3 4 5 8 11 16 21]
W58	[1 9 10 11 12 17 20 21 22]

Tabla 6: Electrodo por usuario antiguo

SELECCIÓN ELECTRODOS	
D10	[1 2 3 4 6 20]
D14	[5 7 9 10]
D15	[1 2 3 6 9]
D16	[1 2 3 4 6 7 20]
D17	[1 2 3 4 5 6 7 8 9 10]
D20	[1 2 3 4 20 21]

Como podemos observar en las tablas anteriores, con este nuevo método se recogen más electrodos que nos aportan una mayor información para poder obtener la muestra de los usuarios. Se puede comprobar de la tabla 6, que la mayoría de los electrodos que se repiten corresponden a los electrodos 1 2 3 4, que se encuentran en la zona central de la cabeza, por lo tanto son los que más información nos aportan.

Finalmente, como sexta configuración tendremos los tiempos de las IMUs, en esta configuración elegimos entre una configuración manual o automática de los tiempos de las IMUs, en esta parte no se ha realizado una comparación entre los dos diferentes tipos, ya que solo emplearemos la detección automática.

Tabla 7: Configuración utilizada

Conf 1	Conf 2	Conf 3	Conf 4	Conf 5	Conf 6
1	5	21	A	1:22	A

Por lo tanto, los datos de la tabla 7 corresponden a los valores que le asignamos en Matlab a cada configuración, para la conf 1 le damos el valor 1 que significa que usaremos el clasificador de tipo LDA. Para la conf 2 le damos el valor de 5, debido a que esta configuración representa las probabilidades a priori que le asignamos, por lo que le damos el valor de 5.

Para la conf 3, le damos el valor de 21, ya que representa las 21 características que le damos a nuestro código. En la conf 4 la A significa automático, ya que nuestra selección de muestra se hace en modo automático.

Para la conf 5 se seleccionan los 22 mejores electrodos que se encuentran y más tarde para realizar el proceso de selección automática de los 27 mejores. Y la conf 6 representamos con la A la selección automática de detección de IMUs.

4.4 Resultados pruebas offline

Una vez aplicado el nuevo código a los archivos obtenidos durante las primeras pruebas, comenzamos a estudiar y obtener los resultados y tablas de los primeros tres usuarios usados. En estos resultados se obtendrán cinco distintas columnas, una primera con la K que es la moda, una segunda con los falsos

positivos por minuto (FP/min), que nos traslada el valor de falsos láseres que hemos ido obteniendo durante el proceso y para cada moda, a continuación, obtenemos el total de láseres emitidos, el total de láseres detectados y el porcentaje de estos TP (%).

Estos códigos han sido aplicados para unas frecuencias diferentes, los valores son los de 0.2 y 0.1, estos dos valores hacen referencia a la frecuencia con la que el código va obteniendo datos de las pruebas. Por lo tanto, habrá que estudiar y llegar a una conclusión de cuál de los dos conviene más usar.

Primero comprobaremos mediante una configuración 9/1, los resultados que obtenemos de los 3 usuarios utilizados para estas pruebas, y los compararemos con los resultados obtenidos en anteriores investigaciones [10] para la misma configuración, además de comparar las dif_media y las muestras obtenidas. Utilizaremos la mejor K que en trabajos anteriores seleccionaron como la que nos arrojaba mejores valores como es la K=5, posteriormente usaremos todos los valores de K para realizar las comparaciones.

Tabla 8: Resultados 9/1 usuarios antiguos y usuarios nuevos.

USUARIOS	dif_media	muestra	K=5	
			FP/min	TP (%)
D10	1.85	559	1.7391	61.53
D17	1.62	159	6.82	95.86
D20	1.90	628	4.37	75
D11	2.00	395	5.23	61.53
D12	1.57	497	7.82	61.53
D13	2.12	863	2.36	45.23
D15	1.62	602	15.88	30.76
D16	1.88	657	1.00	69.23
W26	1.46	300	4.032	10
W30	1.67	265	2.033	20
W33	1.48	310	2.226	0

Tabla 9: Resultados usuarios nuevos offline 5/5

CODIGO USUARIO	FRECUENCIA		K1	K2	K3	K4	K5	RELACION
W26	0.1	TP (%)	20	12	10	8	8	11.6
		FP/min	0.81	1	1	1	1	0.9616
W26	0.2	TP (%)	14	12	8	4	0	7.6
		FP/min	1	1	0.61	0.407	0	0.6034
W30	0.1	TP (%)	2	0	0	0	0	0.4
		FP/min	0.88	0.66	0.44	0.22	0.22	0.484
W30	0.2	TP (%)	0	0	0	0	0	0
		FP/min	0.667	0.22	0	0	0	0.1774
W33	0.1	TP (%)	16.22	8.16	8.16	6.12	4.08	8.44
		FP/min	4.08	2.04	1.22	0.642	0.642	1.725
W33	0.2	TP (%)	10.2	8.16	8.16	6.12	0	6.44
		FP/min	3.052	1.036	0.42	0.222	0	0.946

Como podemos observar dentro de la tabla, los resultados de las pruebas con los nuevos equipos no han sido lo esperado ya que, si bien hemos obtenido un número de FP/min muy bajo, lo cual es bastante significativo, no hemos conseguido obtener una buena detección de los láseres, ya que los porcentajes de obtención son muy bajos. Vamos a comparar los resultados obtenidos con los que se obtuvieron en las investigaciones pasadas por mis compañeros.

Esta comparación se ha hecho con una configuración 5/5 debido a que necesitábamos reducir los archivos de entrenamiento y aumentas los que usábamos para el pseudo.



Tabla 10: Resultados usuarios antiguos offline 5/5 (cambiar frecuencia por

CODIGO USUARIO	FRECUENCIA		K1	K2	K3	K4	K5
D10	0.1	TP (%)	49,23	41,53	35,38	26,15	15,38
		FP/min	6,70	4,63	2,13	1,0647	7,016
D10	0.2	TP (%)	46,15	33,83	13,84	1,53	0
		FP/min	4,97	2,12	0,7	0,52	0
D17	0.1	TP (%)	50,91	95,38	95,38	95,38	89,2
		FP/min	12,58	5,03	3,35	2,86	2,19
D17	0.2	TP (%)	23,73	93,84	85,19	68,09	33,82
		FP/min	13,76	3,35	2,52	1,58	0
D11	0.1	TP (%)	63,15	55,30	44,63	35,65	27,88
		FP/min	13,99	6,74	4,47	3,44	2,74
D11	0.2	TP (%)	58,63	38,56	29,31	13,76	3,07
		FP/min	9,14	4,12	2,74	0,85	0,52
D20	0.1	TP (%)	68,84	54,22	26,28	9,86	6,66
		FP/min	8,67	3,18	1,05	0,348	0,17
D20	0.2	TP (%)	63,97	22,94	6,55	0	0
		FP/min	6,01	1,05	0	0	0
D12	0.1	TP (%)	36,92	27,68	18,45	6,15	1,54
		FP/min	12,68	7,04	4,46	2,27	0,35
D12	0.2	TP (%)	29,22	19,99	1,53	0	0
		FP/min	9,33	3,86	0,35	0	0
D13	0.1	TP (%)	68,84	55,73	52,45	36,06	21,31
		FP/min	9,43	4,88	2,78	2,44	2,44
D13	0.2	TP (%)	57,56	45,12	18,71	1,53	0
		FP/min	6,8	2,97	2,07	0,34	0
D15	0.1	TP (%)	49,23	24,61	15,38	7,69	3,076
		FP/min	16,15	4,74	3,50	1,4	0,35
D15	0.2	TP (%)	36,9	13,84	3,07	0	0
		FP/min	10	3,34	0,17	0	0
D16	0.1	TP (%)	58,45	46,14	23,07	10,76	4,61
		FP/min	4,62	3,26	1,37	0,85	0,68
D16	0.2	TP (%)	52,3	18,45	1,53	0	0
		FP/min	4,11	1,03	0,172	0	0

avance)

Se observa que para los usuarios antiguos se obtuvieron una mejor cantidad de resultados, tanto para porcentaje de láseres detectados como para falsos positivos por minuto, también se obtienen mejores resultados para cualquiera de los dos valores (0.2 o 0.1) que usemos para avanzar dentro de la ventana de análisis.

Aunque se ve una bajada en los FP/min, lo cual es bastante importante, empeora notablemente la cantidad de láseres detectados por láseres emitidos por lo que esto hace que los resultados sean peores.

La tabla, como se indica es para una configuración 5/5, ahora probamos para el usuario W26, el cual nos da los mejores resultados una configuración 7/3 y 2/8 para ver cuáles son los resultados y como varían.

Tabla 11: Resultados W26 configuración 7/3

CODIGO USUARIO	FRECUENCIA		K1	K2	K3	K4	K5
W26	0.1	TP (%)	23.3	10	3.33	0	0
		FP/min	3.31	1.33	0.67	0.67	0
W26	0.2	TP (%)	13.3	3.3	0	0	0
		FP/min	2.33	1.0	0	0	0

Tabla 12: Resultados W26 configuración 2/8

CODIGO USUARIO	FRECUENCIA		K1	K2	K3	K4	K5
W26	0.1	TP (%)	44.8	38.5	35.8	32.0	29.5
		FP/min	11.9	5.9	4.4	3.6	3.1
W26	0.2	TP (%)	39.7	35.8	26.4	10	6.1
		FP/min	8.6	4.4	3.3	1.7	0.4

Lo que nos arrojan estos resultados es que cuando analizamos una mayor cantidad de archivos en el pseudo online obtenemos unos resultados en relación con el número de láseres detectados ampliamente mejor que si

escogemos unos números más bajos para analizar, pero a su vez el número de falsos positivos aumenta bastante. Por lo que realmente no obtenemos una mejoría muy grande.

Para poder comprobar que los cambios realizados en los códigos han sido los correctos y no se ha introducido ninguna línea incorrecta, se ha hecho una comprobación mediante datos obtenidos previamente en pasadas investigaciones.

El proceso para realizar esto ha sido cargar usuarios antiguos y aplicarle el nuevo código que hemos hecho. Para que estos nuevos usuarios puedan funcionar en nuestro nuevo código ha sido necesario crear una función de conversión, explicada en el punto 3.5.

Una vez obtenemos los resultados se pueden comparar que son exactamente los mismos que obteníamos con el código antiguo, por lo que no se ha cometido ningún fallo a la hora de la adaptación del código antiguo.

Tabla 13: Resultados comparación códigos

Código		K1	K2	K3	K4	K5
Antiguo	TP (%)	49.23	41.53	35.38	26.15	15.38
	FP/min	6.7	4.63	2.13	1.0647	7.016
Nuevo	TP (%)	49.23	41.53	35.38	26.15	15.38
	FP/min	6.7	4.63	2.13	1.0647	7.016

Como también se puede observar de los resultados offline de la tabla xx, se obtiene un mejor resultado para un valor de 0.1 que para un valor de 0.2, por lo tanto, nos quedaremos con el uso de dicho valor.

4.5 Realización pruebas online

La segunda parte y última de nuestra investigación consiste en la realización de una serie de pruebas online a unos determinados usuarios. Como comento en el punto 3.6, se ha llevado a cabo un protocolo diferente al usado para las pruebas offline.

Este consiste en realizar una primera de serie de pruebas en modo offline, con estas pruebas realizadas obtendremos los valores de la muestra y de la dif_media, para luego poder generar el modelo que más tarde cargaremos en la arquitectura online.

Esta arquitectura se ejecutará, y tendremos que intentar detectar la parada del usuario ante la emisión del láser antes de que se genere.

La realización de estas pruebas ha llevado un gran trabajo de coordinación con el investigador Vicente Quiles, ya que ha tenido que adaptar gran parte de la arquitectura al modo online. Para poder realizar parte de esta coordinación ha sido necesario implementar parte de los códigos que comento en el punto 3.5, como son los códigos en los que agrupamos los clasificadores, filtros y características.

Estos códigos son importantes comprimirllos cada uno en su script concreto, ya que se introducen por bloques dentro de la arquitectura online y así no se realizan más acciones.

Para estas pruebas en modo online se han usado exactamente los mismos materiales que se mencionan para las pruebas offline, como son la cinta y el láser, el gorro de electrodos, las IMUs y el equipo de medición de EEG.

4.5 Resultados pruebas online

Una vez realizado el protocolo sobre los usuarios que hemos seleccionado para realizar las pruebas en modo online, procedemos a obtener los resultados que nos arrojan mediante tablas y gráficas.

Dentro de la realización de estas pruebas no hemos obtenido los resultados esperados, ya que no hemos conseguido detectar la parada del usuario para el primer voluntario, y con los 2 últimos usuarios detectamos la parada demasiado pronto por lo que no dejaba tiempo al láser a emitir. A pesar de que los valores de dif_media y muestra son correctos. Y la selección de electrodos se ha realizado con coherencia.

Tabla 14: Resultados dif_media, muestra y electrodos usuarios modo online.

CÓDIGO USUARIO	dif_media	Muestra	Muestra (s)	Selección electrodos
W55	1.60	240	0.48	[1 2 5 6 13 16 18]
W56	1.41	300	0.6	[1 3 4 5 8 11 16 21]
W58	1.53	320	0.64	[1 9 10 11 12 17 20 21]

Primero obtendremos los resultados para una configuración 4+1, en la que usamos 4 ficheros de entrenamiento y 1 para aplicarle el pseudo, para así poder ver los resultados que hemos obtenido.

Tabla 15: Resultados configuración 4+1 modo offline

CÓDIGO USUARIO		K1	K2	K3	K4	K5
W55	FP/min	4.17	1.04	0	0	0
	TP (%)	0	0	0	0	0
W56	FP/min	3.96	0.99	0.99	0	0
	TP (%)	10	0	0	0	0
W58	FP/min	1.058	0	0	0	0
	TP (%)	0	0	0	0	0

Como se puede observar de los resultados obtenidos, se deduce que no han sido los esperados. En parte estos resultados de la tabla han salido tan malos debido a que se necesitarían más pruebas tanto de entrenamiento como para aplicarle el pseudo, por lo que se obtienen esos malos resultados. Pero como para generar la tabla hemos utilizado únicamente los archivos que realizamos primero de modo offline y solo se hacen cinco para generar el modelo.

Para los segundos usuarios intentamos subir el valor de la moda (k) para poder retrasar la detección de la intención de parada por parte del usuario, pero aun así no se consiguió retrasarla y nada más comenzar la tarea del láser (602), se detectaba la intención de parada y se finalizaba la prueba.

Como podemos observar para el primer usuario que realizó estas pruebas, comprobamos mediante la gráfica azul las 4 tareas diferentes como son la 400, 500, 601 y 602 y mediante la gráfica roja podemos ver como cuando comienza la tarea del láser (602), se emiten los 10 láseres seguidos para más tarde el láser de desactivación.

Como se comenta en el protocolo online del punto 3.7, en la tarea 400 se comienza a ejecutar el Código haciendo una calibración del usuario y ejecutando códigos, a continuación se inicia la tarea 500 que es la cual nos permite poder arrancar la cinta de correr mediante un comando desde Matlab, la siguiente tarea es la 601, en esta tarea se arranca la cinta y establecemos un

período de arranque en el cual el usuario se acostumbra al arranque de la cinta y se estabilice caminando, la última tarea es en la cual se encuentran los láseres y se realizará el comando de parada de cinta por parte de nuestra arquitectura, en la siguiente figura se puede observar los tramos.

Además, en rojo se diferencian la activación del láser para el primer usuario (W55). Cuando el valor es de 100 es cuando se activa el láser y cuando el valor sube hasta los 200 es cuando el láser se desactiva.

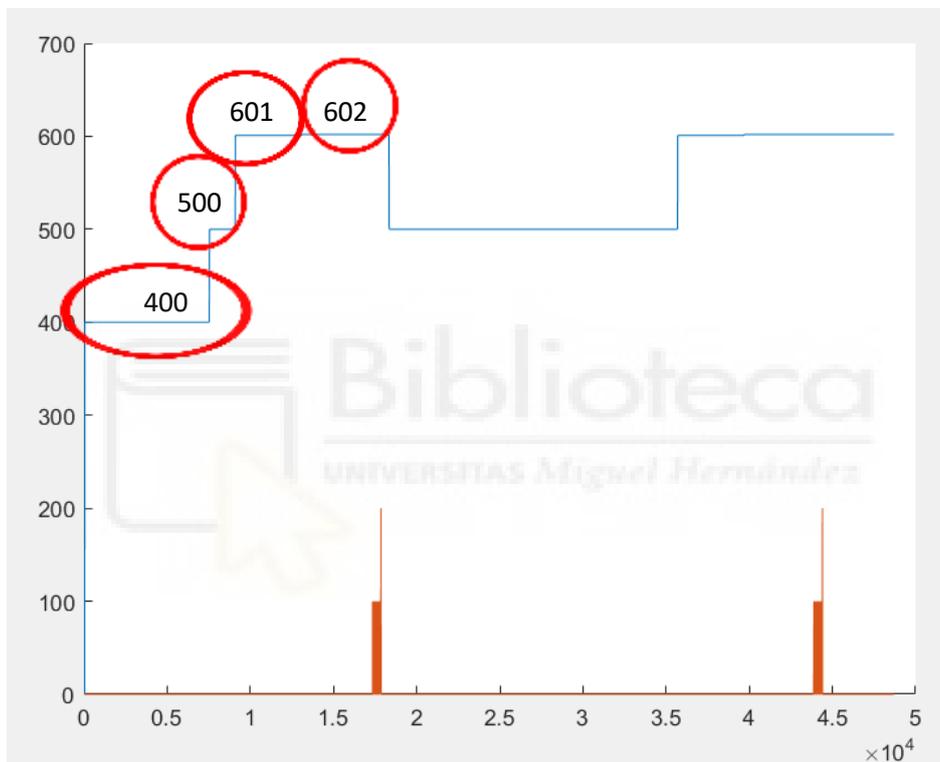


Figura 25: Resultados usuario W55.

Este proceso se realizó 8 veces, por lo que se obtuvieron 16 repeticiones, debido a que cada vez que se hacía se activaba el láser 2 veces.

El problema que se nos generó con este usuario es que todas y cada una de las repeticiones que se realizaron daban los mismos resultados y daban como resultado los mismos valores. El usuario seguía realizando su curso de caminata hasta la aparición del láser, luego se detenía y ya la cinta realizaba la tarea de stop.

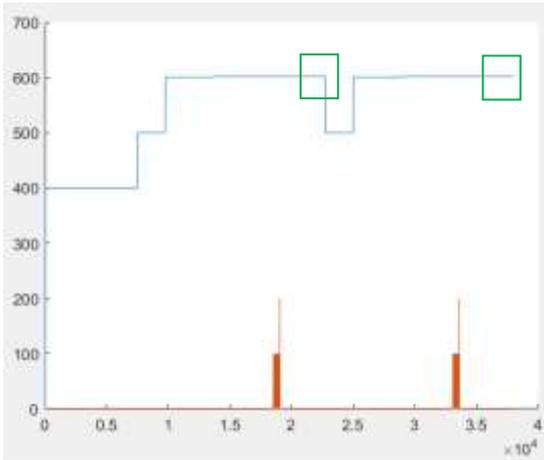


Figura 26: Resultados primera repetición W55

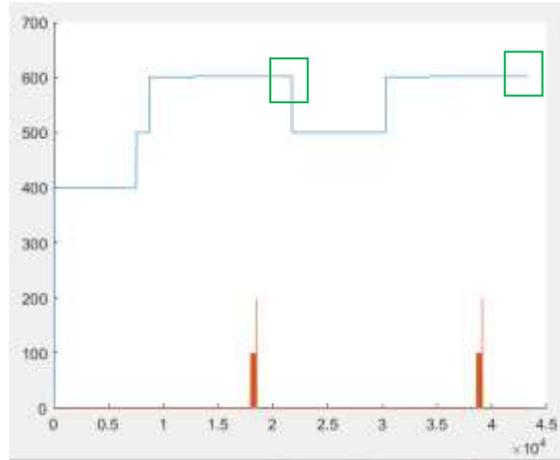


Figura 27: Resultados segunda repetición W55

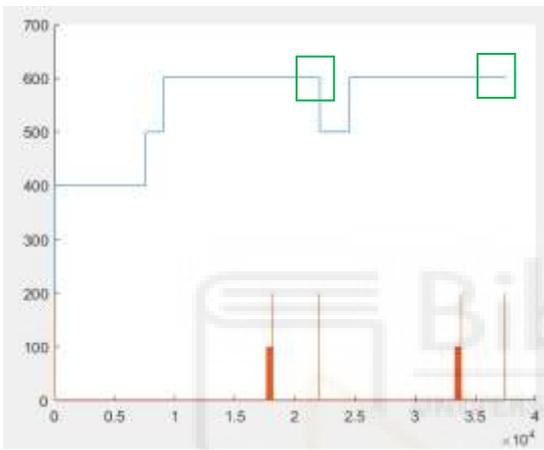


Figura 28: Resultados tercera repetición W55

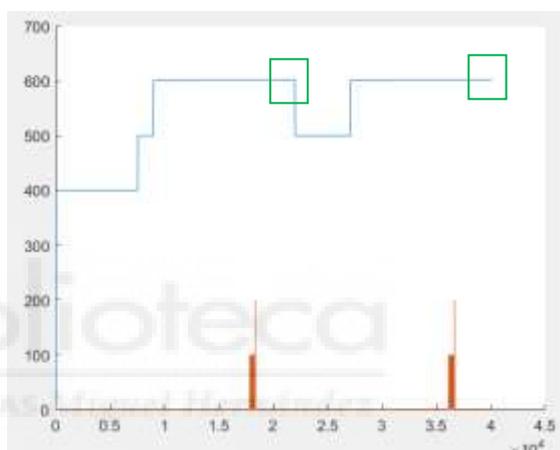


Figura 29: Resultados cuarta repetición W55

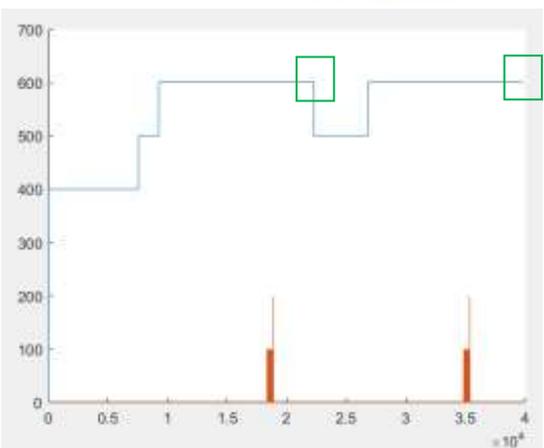


Figura 30: Resultados quinta repetición W55

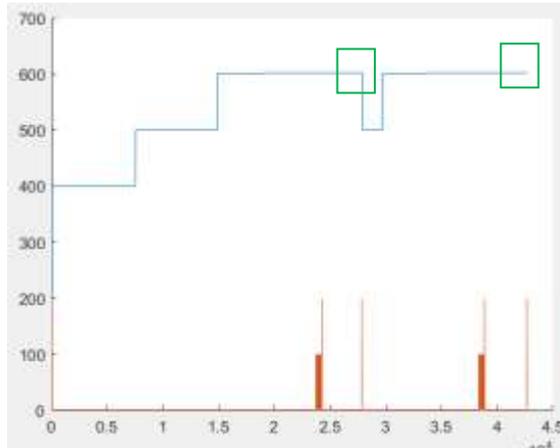


Figura 31: Resultados sexta repetición W55

Como se puede ver en las figuras 26, 27, 28, 29, 30 y 31, los resultados obtenidos son muy parecidos, en los que solo varían los tiempos de emisión de láser y de alguna tarea, pero la cantidad de láseres y la secuencia que se usa a lo largo de todas las repeticiones es básicamente la misma.

Este caso que ocurre con el usuario W55, significa que nunca detectaba en evento de parada por el usuario por lo que esto nos generaría unos bajos FP/min, pero a su vez unos muy bajos TP (%), por lo que los resultados no serían los correctos.

Mediante los rectángulos verdes se puede observar el punto en el que se emite el comando de detección de la cinta, que eso para cada repetición se originó en un determinado tiempo.

Para los segundos usuarios obtuvimos completamente los resultados, al contrario, en vez de que nunca detectara el evento y mandara el comando de parada, esta vez lo detectaba siempre el evento y detenía la cinta, por lo que este caso es bastante peor que el primero, siendo los dos nada buenos para obtener resultados significantes.

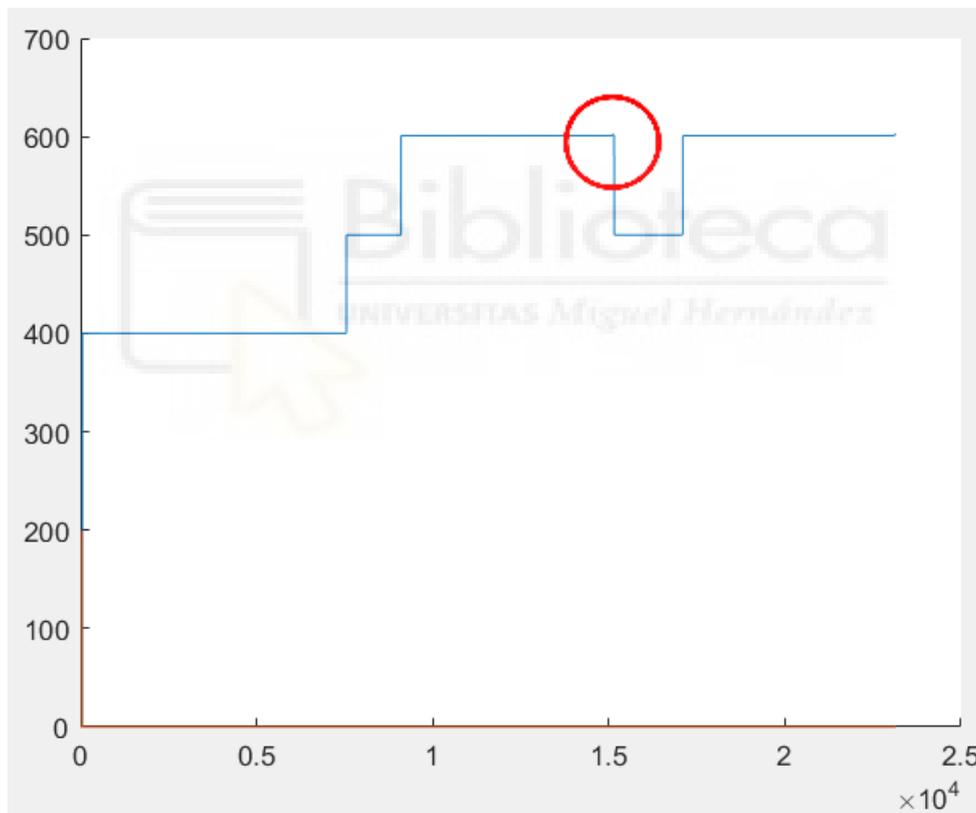


Figura 32: Resultados usuario W58

En esta segunda gráfica se pueden ver los resultados para el usuario W58, con este usuario y con el usuario W56, nos da un resultado completamente distinto. En este podemos ver como nada más llegar la gráfica azul al valor de 602, marcado por el círculo rojo, la cinta se detiene y pasa a la tarea 500, por lo que no deja tiempo al láser a emitir y que el usuario pueda detectarlo o no.

Esto al igual que para el usuario W56, se repite en todas las repeticiones que hemos realizado, como se puede ver en las siguientes gráficas.

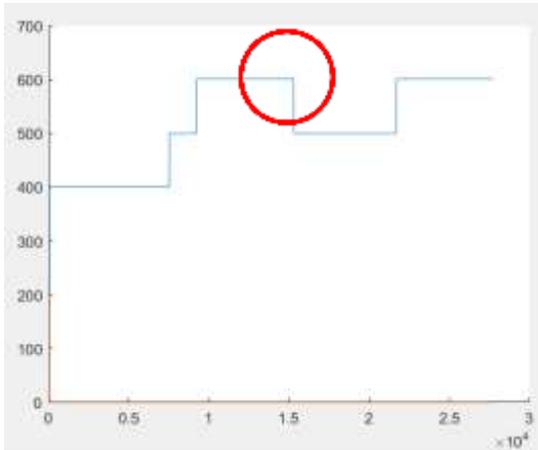


Figura 33: Resultados primera repetición W56

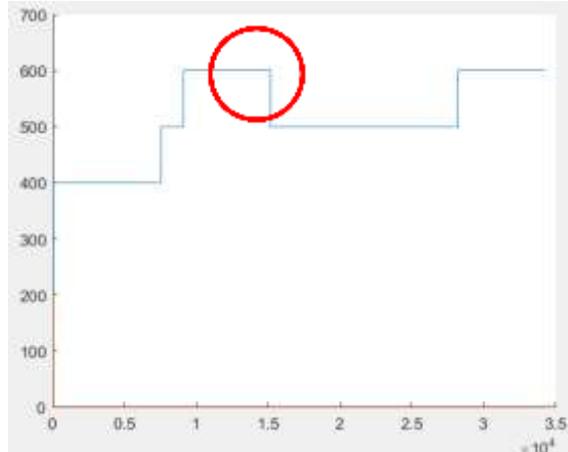


Figura 34: Resultados segunda repetición W56.

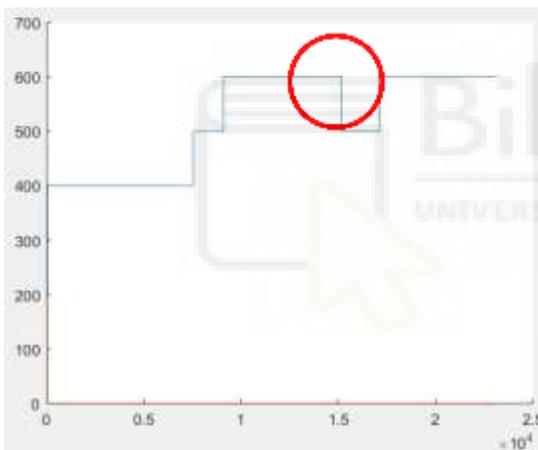


Figura 35: Resultados tercera repetición W56.

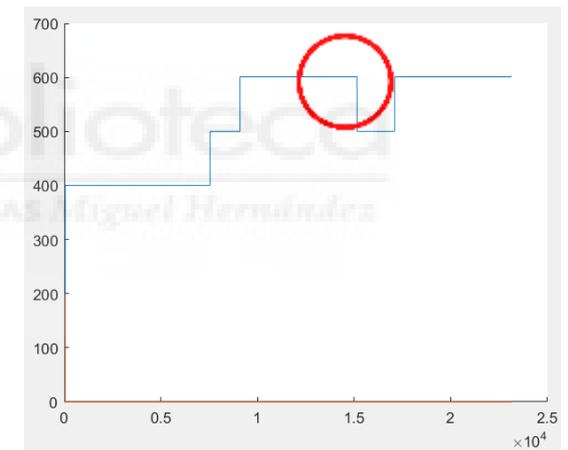


Figura 36: Resultados cuarta repetición W56.

Como se puede ver en las gráficas, los resultados son prácticamente iguales, en la línea amarilla de abajo se observa que el láser no consigue emitirse en ninguna de las repeticiones, debido a que nada más comenzar la tarea 602, la tarea de la emisión del láser detectaba la parada y mandaba la orden de detenerse y la cinta procedía a pararse, por lo que no hemos conseguido nuestro objetivo.

En el punto señalado por el círculo rojo, se observa que durante un instante se genera la tarea 602, la tarea el láser, para más tarde volver a la tarea 500, la cinta se para y se termina la repetición.

En cuanto a la detección que registramos mediante las IMUs se corresponden con las siguientes gráficas, en las siguientes gráficas se observa que durante la evolución de la línea azul hay dos zonas en los que oscila, por lo que se corresponde con la parada de la cinta y posterior parada del usuario.

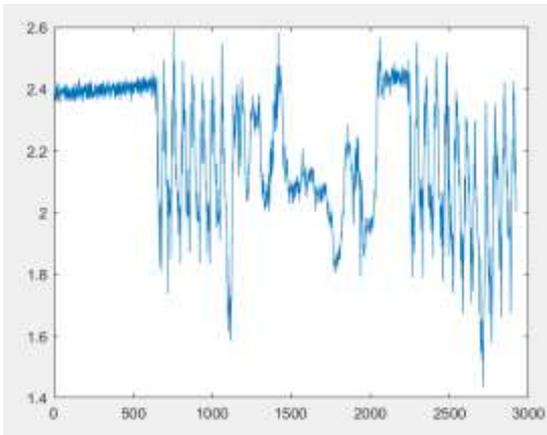


Figura 37: Resultados IMUs W58

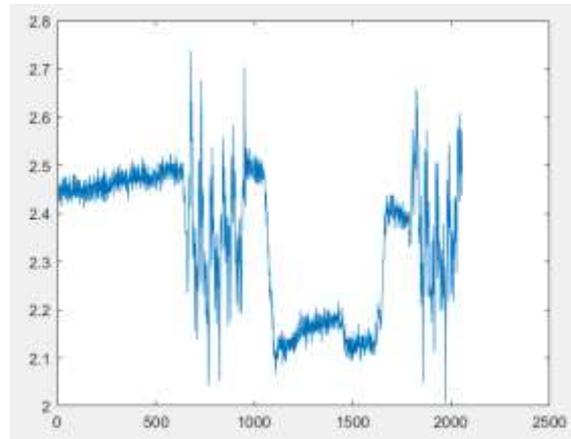


Figura 38: Resultados IMUs W55

En ambas gráficas se muestra la evolución de las detecciones de las paradas de las IMUs, en la de la izquierda se observa cómo solo dura 2100 muestras, que dividido entre la frecuencia de las IMUs la cual es de 30 Hz, se nos queda un tiempo de 70 s que es el tiempo que duró cada repetición para el usuario W58, por lo que se puede observar cuando hacia las dos paradas.

En cambio, en la figura de la derecha se puede observar que corresponde al primer usuario, que era el que se nos emitía el láser constantemente y no conseguíamos detectar la intención de parada. Esta detección dura más debido a que en estas pruebas sí que se emitía el láser por lo que el usuario caminaba más sobre la cinta sin detección, por lo que estamos hablando de que dura aproximadamente 3000 muestras entre la frecuencia de las IMUs se queda en un tiempo de 100s justo el doble de lo que duraba para el usuario W58.

Por lo tanto, si se envía el comando a la cinta antes de que el láser se active se consideraría una activación incorrecta, en cambio si aparece el láser y el comando se envía antes de que el usuario se detenga se considera una detección correcta, pero si parase la cinta justo antes de detectar que se detiene el usuario sería incorrecto puesto que el comando se ha mandado tarde.

En nuestro caso, en el primer usuario (W55), no se detectaba la intención de parada por parte del usuario hasta que no se mandaba la orden de activación del láser y el usuario realizaba la parada correspondiente. Para los dos

siguientes usuarios, se detectaba la intención de parada nada más comenzar la tarea lo cual es erróneo y no nos aporta ningún resultado favorable.

Para poder mejorar este resultado habría que optimizar los códigos más, reduciendo tiempos de espera y mejorando los scripts hasta el punto en el que se pueda detectar bien la intención de detenerse por parte del usuario.





5. Conclusión

En este punto evaluaremos todos los resultados que hemos ido obteniendo a lo largo de la investigación, además de exponer posibles mejoras para futuros trabajos.

Para comenzar podemos hablar de la inclusión de la función `laser_vector` dentro de la arquitectura, esta función ha cumplido su función y se han obtenido los resultados esperados, aparte esta función se puede manipular y variar a antojo del investigador para poder adaptarlo a las necesidades del proyecto que se estuviera llevando a cabo.

En segundo lugar, comentaremos los primeros resultados obtenidos en el modo offline con el nuevo equipo utilizado. Estos resultados no han sido los esperados, debido a que se pretendía obtener una mejora con respecto a los obtenidos en investigaciones anteriores. Pero los resultados arrojados no son del todo correctos, aunque hayamos hecho una gran mejoría en la reducción de los FP/min, se ha reducido en una gran medida los resultados para el TP (%). Se han realizado pruebas para ver si las adaptaciones del código no eran las correctas, pero se obtuvieron los mismos resultados, por lo que se puede entender que las mejoras en el equipo y en la forma de medir no han sido mejoradas.

Seguidamente podemos hablar de los resultados de parte de los códigos añadidos a la serie de scripts que se habían hecho en anteriores investigaciones. Estos resultados han sido buenos, ya que se ha conseguido obtener los resultados esperados para cada uno de los códigos que se han escrito para este trabajo.

Continuamos con los resultados que se han obtenido para las pruebas realizadas en modo online. Los resultados de estas pruebas tampoco han sido los esperados, debido a que no se ha conseguido el objetivo previamente establecido.

Aunque se han realizado a 3 diferentes sujetos las pruebas, con ninguno de ellos se ha podido acercar a los resultados que nos esperábamos, esto posiblemente haya sido problema de la adaptación del nuevo equipo utilizado a esta arquitectura o viceversa. Cabría poder resolver esta problemática para poder esperar nuevos resultados más precisos.

En la vista de todo lo sucedido durante este proyecto de investigación y a los resultados no favorables obtenidos, se puede deducir que existen muchas mejoras a realizar para futuras investigaciones de cara a poder llegar al punto deseado.

Una de las posibles mejoras que se pueden acometer es a la hora de la optimización del código que genera el modelo y posteriormente el pseudo-online, ya que este código contempla muchas variables y tarda mucho en procesarse, una posible optimización de este código puede que realice el trabajo más simple y rápido.

Otro futuro posible trabajo es la ampliación de este mismo al uso del exoesqueleto, una vez mejorado los resultados obtenidos en cinta de correr, para poder así avanzar en el campo y obtener una visión más amplia.





Bibliografía

- [1] Hankey, G. J., & Warlow, C. P. (1999). Treatment and secondary prevention of stroke: evidence, costs, and effects on individuals and populations. *The Lancet*, 354(9188), 1457-1463.
- [2] Hernández, M. G., Cruz, S. A. D., Rubiera, A. T., & Pérez, M. S. (2017). Epidemiología de la enfermedad cerebrovascular en un área intensiva municipal. *Revista Cubana de Medicina Intensiva y Emergencias*, 16(2), 69-79.
- [3] Alcázar-Fabra, M. (2012). El potencial de acción. *MoleQla: revista de Ciencias de la Universidad Pablo de Olavide*, (5), 166-170.
- [4] Guillén Lladó, José M^a (2016). Ondas cerebrales, estrés y biomagnetismos emocional. Sant Pau Centro Terapèutic.
- [5] Elvira, M., Iáñez, E., Quiles, V., Ortiz, M., & Azorín, J. M. (2019). Nuevos métodos para la detección de obstáculos inesperados durante la marcha normal a través de señales EEG. In *XI Jornadas de Automática* (pp. 55-62). Universidade da Coruña, Servizo de Publicacións.
- [6] Galindo Merchán, Daniel (2011). Introducción a los sistemas brain computer interface. <http://www.lacofa.es/general/introduccion-a-los-sistemas-brain-computer-interface>
- [7] Medina, B., SIERRA, J. E., & ULLOA, A. B. (2018). Técnicas de extracción de características de señales EEG en la imaginación de movimiento para sistemas BCI. *Revista ESPACIOS*, 39(22).
- [8] Santamaría Vázquez, E. (2016). Diseño y desarrollo de una aplicación para controlar un teléfono móvil mediante sistemas Brain Computer Interface (BCI) orientada a personas con grave discapacidad.
- [9] Hornero, R., Corralejo, R., & Álvarez, D. (2012). Brain-Computer Interface (BCI) aplicado al entrenamiento cognitivo y control domótico para prevenir los efectos del envejecimiento. *Fundación General CSIC LYCHNOS [en línea]*, 8
- [10] Fontelles Congost, R. (2021). Mejoras en la detección de la aparición inesperada de un obstáculo durante la marcha mediante señales cerebrales. Universidad Miguel Hernández, Elche.

- [11] María Elvira Giner (2019). Detección de obstáculos durante la marcha a partir de señales EEG. Universidad Miguel Hernández, Elche.
- [12] Salazar-Varas, R., Costa, Á., Iáñez, E., Úbeda, A., Hortal, E., & Azorín, J. M. (2015). Analyzing EEG signals to detect unexpected obstacles during walking. *Journal of neuroengineering and rehabilitation*, 12(1), 1-15.
- [13] Hortal, E., Planelles, D., Iáñez, E., Costa, A., Úbeda, A., & Azorín, J. M. (2016). Detection of gait initiation through a ERD-based brain-computer interface. In *Advances in Neurotechnology, Electronics and Informatics* (pp. 141-150). Springer, Cham.

