

**MÁSTER UNIVERSITARIO EN
SISTEMAS ELECTRÓNICOS AVANZADOS**

TRABAJO FIN DE MÁSTER

***DESARROLLO E INTEGRACIÓN DE SISTEMAS
BRAIN-TO-VEHICLE PARA EL ESTUDIO DE
RESPUESTAS EMOCIONALES EN UN SIMULADOR
DE CARRETERA***

Estudiante *Gallego Gimeno, Alba*
Director *Basterretxea Oyarzabal, Koldo*
Departamento *Tecnología Electrónica*
Curso académico *2021-2022*

Bilbao, 26, junio, 2022



DESARROLLO E INTEGRACIÓN DE SISTEMAS BRAIN-TO-VEHICLE PARA EL ESTUDIO DE
RESPUESTAS EMOCIONALES EN UN SIMULADOR DE CARRETERA

RESUMEN

La mayoría de personas utilizan a diario medios de transporte para desplazarse, entre ellos, el más común es el coche. Esto, además de aumentar el riesgo a sufrir un accidente, supone que las personas pasan gran parte de su tiempo en su coche, por lo que conseguir un confort en el desarrollo de nuevos vehículos es un factor a tener en cuenta.

La gran mayoría de los accidentes de tráfico se producen a causa del factor humano, dentro de estos, gran parte son debido a distracciones, por lo que el conductor resulta un componente principal para garantizar una buena seguridad en la conducción. De esta forma, la distracción debida a los estados emocionales del conductor afecta directamente a los procesos cognitivos requeridos para realizar correctamente otras tareas como conducir.

En este proyecto se pretende desarrollar un sistema de monitorización cerebral por electroencefalograma (EEG) orientado a la conducción capaz de detectar los diferentes estados emocionales del conductor y activar diferentes configuraciones en el modo de conducción en función de estos. Con esto se pretende mejorar el confort y la seguridad del conductor en la medida de lo posible, adaptando algunas características de la dinámica del vehículo como la suspensión.

Para esto, se va a desarrollar un sistema Brain Computer Interface (BCI) donde el primer paso será la adquisición de las señales cerebrales y su procesamiento. Se realizará una extracción de las características relevantes de estas y, por último, se clasificarán para proporcionar una salida indicando el estado emocional detectado. Esta salida proporcionada por el sistema, se introducirá en un simulador de conducción del que dispone la empresa AIC donde se ajustarán los parámetros de la dinámica vehicular pertinentes.

Tras realizar el desarrollo y análisis del sistema siguiendo los pasos especificados en el artículo [1], se han obtenido resultados de precisión en la clasificación cerca de los valores de ejemplo similares a los del artículo.

Como conclusión, cabe destacar que pese a ser un sistema con una aceptable precisión con la que trabajar, si este se quisiera implementar en la vida real, sería necesario aumentar la precisión para obtener más fiabilidad del sistema.

Palabras Clave: EEG, BCI, Reconocimiento Emociones, Vehículos Inteligentes, Inteligencia Artificial.

eman ta zabal zazu



Universidad
del País Vasco

Euskal Herriko
Unibertsitatea

BILBOKO
INGENIARITZA
ESKOLA



ÍNDICE

Capítulo 1. INTRODUCCIÓN.....	5
1.1. Motivación y contexto	7
1.2. Objetivos.....	9
1.3. Alcance y beneficios.....	9
1.4. Resumen de contenido.....	10
Capítulo 2. ESTADO DEL ARTE.....	11
2.1. Vehículos autónomos	11
2.2. Las emociones y el cerebro	11
2.2.1 Qué son las emociones	11
2.2.2. Las emociones en el cerebro	12
2.3. Reconocimiento de emociones	13
2.3.1. Neurociencia.....	13
2.3.2. Tipos de interfaces cerebro-computadora	14
2.4. Brain Computer Interface (BCI).....	15
2.5. Confort vehicular.....	16
Capítulo 3. DEFINICIÓN DE LA SOLUCIÓN PROPUESTA.....	18
3.1. Funcionamiento	18
3.2. Diseño general	18
3.3. Especificaciones del sistema	19
3.3.1. Tipo de interfaz cerebro-computadora	19
3.3.2. Características a extraer de las señales EEG.....	21
3.3.2.1. Transformada EMD (Descomposición Empírica en Modos)	21
3.3.2.2. Primera diferencia de fase.....	22
3.3.2.3. Primera diferencia de series de tiempo	23
3.3.2.4. Energía normalizada	23
3.3.4. Modelos de clasificación emocional	23
3.3.5. Clasificadores	25
3.3.6. Base de datos.....	26
3.3.7. Protocolos de actuación.....	27
3.3.8. Protocolos de comunicación con la plataforma de simulación	28



3.3.9. Software.....	29
Capítulo 4. METODOLOGÍA.....	34
4.1. Definición y distribución de tareas.....	31
4.2. Estudio bibliográfico	32
4.3. Definición del código	32
4.4. Desarrollo del código.....	36
4.4.1. Cálculo de características	36
4.4.2. Extracción de las características de las señales de DEAP dataset.....	38
4.4.3. Generación de las matrices de datos a partir de las señales	38
4.4.4. Entrenamiento del clasificador.....	39
4.5. Pruebas experimentales y análisis de resultados	40
4.6. Desarrollo de protocolos de actuación	5
4.7. Validación y test	7
4.8. Implementación hardware	8
Capítulo 5. EJEMPLO DE APLICACIÓN.....	58
Capítulo 6. CONCLUSIONES Y TRABAJO FUTURO.....	59
BIBLIOGRAFÍA.....	61
Anexos.....	65



ÍNDICE DE FIGURAS

Figura 1. Plataforma de simulación AIC [5]	8
Figura 2. El sistema límbico [9]	12
Figura 3. Sistema internacional de posicionamiento de electrodos 10-20 [12].....	13
Figura 4. Aplicación neuromarketing	14
Figura 5. Sistema BCI [18]	16
Figura 6. Rango frecuencial de los efectos de las vibraciones en seres humanos	17
Figura 7. Diseño general del sistema.....	19
Figura 8. Tabla comparativa interfaces cerebro-computadora	19
Figura 9. Tipos de ondas cerebrales [32]	20
Figura 10. Función intrínseca a partir de señal EEG.....	22
Figura 11. Modelo de Russell	24
Figura 12. Modelo de Plutchik.....	24
Figura 13. Base de datos DEAP dataset [41]	26
Figura 14. Contenido fichero data_preprocessed_matlab.zip	27
Figura 15. Protocolos de comunicación con el simulador	28
Figura 20. Esquema metodología	31
Figura 21. Diagrama de Gantt.....	31
Figura 22. Overlapping de señales	33
Figura 23. Canales de las EEG DEAP dataset.....	34
Figura 24. Técnica de undersampling.....	35
Figura 25. Diseño final del sistema.....	36
Figura 26. Señal EEG	36
Figura 30. Señal filtrada con EMD.....	37
Figura 48. Representación matriz de datos.....	40
Figura 49. Análisis PCA matriz de datos	40
Figura 50. Matriz correlación características	1
Figura 51. Análisis de clasificadores para la valencia alta o baja	1
Figura 52. Matriz de confusión de clasificación valencia con PCA	2
Figura 53. Matriz de confusión de clasificación valencia sin PCA.....	2
Figura 54. Análisis de clasificadores para la excitación alta o baja.....	3
Figura 55. Matriz de confusión de clasificación excitación con PCA.....	4
Figura 56. Matriz de confusión de clasificación excitación sin PCA.....	4
Figura 57. Curvas de suspensión modo confort.....	5
Figura 58. Curvas de suspensión modo sport	6
Figura 59. Curvas de suspensión modo personalizado tristeza	6
Figura 60. Curvas de suspensión modo personalizado preocupación	7
Figura 63. Esquema Simulink UDP send.....	8
Figura 64. Esquema Simulink UDP receive.....	9
Figura 66. Test comunicación UDP.....	9
Figura 67. Conductor en una situación de estrés.....	11
Figura 27. Función extracción características	17
Figura 28. Código filtro entrada.....	17
Figura 29. Código transformada EMD de las señales.....	17
Figura 31. Código inventariado señales	17
Figura 32. Código primera diferencia de fase	18
Figura 33. Código primera diferencia de series de tiempo.....	18
Figura 34. Código de carga de las señales originales.....	18
Figura 35. Código energía normalizada	19
Figura 36. Función de extracción señales DEAP	19



<i>Figura 37. Código de carga de los ficheros DEAP</i>	19
<i>Figura 38. Código de carga de las señales de los 8 electrodos</i>	20
<i>Figura 39. Llamada a la función de extracción de características</i>	20
<i>Figura 40. Código extracción de matrices de características de las señales de DEAP</i>	20
<i>Figura 41. Código undersampling matrices de datos</i>	21
<i>Figura 42. Código matriz de datos valencia</i>	21
<i>Figura 43. Código matriz a introducir en el clasificador</i>	21
<i>Figura 44. Menú de la aplicación Classification Learner</i>	21
<i>Figura 45. Opción de entrenar todos los clasificadores</i>	21
<i>Figura 46. Código de uso de los clasificadores</i>	22
<i>Figura 47. Código especificación del tipo de emoción</i>	22
<i>Figura 61. Verificación del sistema</i>	22
<i>Figura 62. Resultado de la verificación</i>	23
<i>Figura 65. Función asignación parámetros de suspensión</i>	23
<i>Figura 16. Dispositivo AIR</i>	24
<i>Figura 17. Dispositivo NEXSTEM HEADSET</i>	25
<i>Figura 18. Dispositivo DSI 7</i>	25
<i>Figura 19. Dispositivo Ultracortex EEG Headset</i>	25

Capítulo 1. INTRODUCCIÓN

1.1. Motivación y contexto

El uso de la inteligencia artificial aplicada a la medicina está en auge, esto se debe a que aporta muchas ventajas a la hora de supervisar a los pacientes o realizar un diagnóstico preciso.

Los límites de la inteligencia artificial cada vez son menos, pues gracias a estas máquinas son capaces de reproducir un comportamiento de modo similar al de un ser humano a través de técnicas como el machine learning. Esto daría posibilidad a que estas fueran capaces de analizar los resultados de diferentes pruebas con el fin de obtener una evaluación más precisa incluso que la humana, tanto es así, que se prevé que en 2025 existan avatares digitales capaces de facilitar la gestión de la salud [2].

Por esto, resulta obvio que las nuevas tecnologías de la inteligencia artificial van a crecer de cara a los próximos años, mejorando así la calidad de vida de las personas. Estas aplicaciones lejos de usarse únicamente en el ámbito de la medicina, podrían ayudarnos a adaptar herramientas de la vida cotidiana para mejorar la seguridad y confort de las personas analizando su estado de salud. Un claro ejemplo de este tipo de aplicaciones podría ser en la conducción, ya que esta es una de las principales causas de muerte o enfermedad desde hace décadas.

La mayoría de personas utilizan a diario un vehículo para desplazarse, donde el más utilizado es el coche. Esto implica que además de pasar gran parte de su tiempo en estos desplazamientos, el riesgo a sufrir un accidente aumenta significativamente al ser de 1 entre 107 la probabilidad de tener un accidente.

Según el informe ‘Cuestiones de Seguridad Vial, Conducción Eficiente, Medio Ambiente y Contaminación’ del Ministerio de Fomento junto con la DGT [3], entre el 70 y 90% de los accidentes de tráfico provienen del factor humano. Además, se estima que el 31% de accidentes totales se deben a distracciones del conductor. Estos altos porcentajes dejan ver que el conductor es la componente principal para garantizar una buena seguridad en la conducción.

Poniendo el foco en los accidentes causados por distracciones, estas pueden ser de cuatro tipos: visuales, cognitivas, físicas o auditivas. Las reacciones emocionales debidas a un determinado suceso o estímulo interfieren directamente con los procesos cognitivos requeridos para realizar correctamente otras tareas, en este caso conducir. A esto se le conoce como distracción emocional. Este tipo de distracción afecta en gran manera al individuo en cualquier tarea que este pueda estar realizando ya que el sistema cognitivo se ve comprometido [4].

Este tipo de sucesos desencadenan reacciones en el conductor perjudiciales en la carretera, ya que como efecto la persona tiende a caer en una distracción de su tarea

principal, lo que resulta en ser una de las situaciones por las que suceden gran cantidad de accidentes en la carretera. Estos sucesos han provocado que numerosas empresas e instituciones se centren en el desarrollo de nuevas tecnologías capaces de prevenir este tipo de accidentes analizando el comportamiento del conductor para poder adelantarse a los actos. Además, el hecho de poder actuar en función del estado cognitivo del conductor implica una mejora destacable el confort de este al hacer posible la adaptación del vehículo en cada uno de sus estados.

Por estos factores mencionados, se ha visto motivada la idea de desarrollar un sistema de monitorización de usuario capaz de analizar el estado emocional del conductor y activar un protocolo de actuación en función de las respuestas obtenidas.

Un ejemplo de empresa que sigue esta línea de investigación en la que cobra gran importancia el confort, se encuentra en la empresa Automotive Intelligence Center (AIC), un centro de competencia mundial especializado en la automoción donde se ha desarrollado el proyecto con el fin de llegar a realizar pruebas experimentales, si el tiempo lo permite, en el simulador de carretera (Figura 1) que se dispone en el departamento del Virtual Development Center (VDC).

Esta plataforma de simulación de conducción tiene un sistema dinámico que consta de seis grados de libertad, con una plataforma inferior y otra superior conectadas mediante seis actuadores eléctricos, frente a un sistema visual compuesto de tres pantallas, y complementado por un sistema de audio. El software de arquitectura modular controla la plataforma, ejecuta los modelos vehiculares y el renderizado de los entornos gráficos [5].

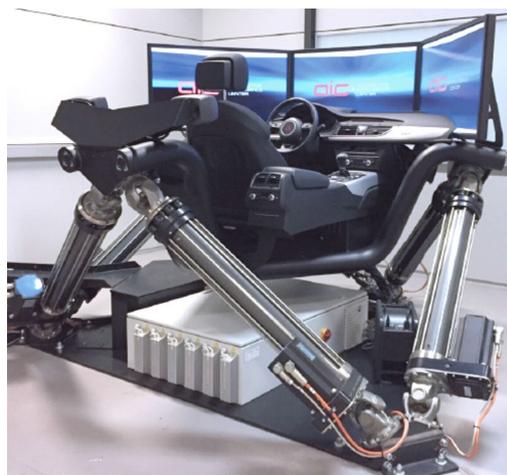


Figura 1. Plataforma de simulación AIC [5]

Este departamento sigue tres áreas de dominio: Modelización de componente, sistema y vehículo, Ride comfort y Handling Performance [5]. Entre estas áreas de investigación, el proyecto estaría alineado con el dominio de Ride comfort.



1.2. Objetivos

Como se ha mencionado anteriormente, se ha desarrollado un sistema de monitorización del conductor usuario para analizar el estado emocional a través de ondas cerebrales. Este sistema medirá las señales de diferentes regiones del cerebro del usuario las procesará y extraerá las características con las que se realizará la clasificación de estados emocionales.

Los principales objetivos del proyecto son la creación y desarrollo de un monitor de ondas cerebrales, intentando utilizar todos los recursos disponibles con el fin de llevar lo más lejos posible el proyecto. Para ello, es preciso realizar un estudio de investigación en el tema a tratar. Además, conocer el funcionamiento de los elementos de los que disponga el software, y, si se adquiere el hardware, el modo de uso y conexión del mismo con el programa.

Una vez se haya realizado una investigación exhaustiva, es decir, se haya adquirido conocimiento sobre las emociones en el cerebro, como adquirir y procesar las señales cerebrales, la extracción de datos y la posterior clasificación, el siguiente objetivo es la elaboración del diseño en el que se va a basar el proyecto.

Cuando se especifique el diseño a realizar y este sea capaz de cubrir las necesidades del proyecto, se procederá a generar el código en MATLAB que dictará el funcionamiento del sistema. Este será el encargado de procesar las señales de entrada, hacer el filtrado correspondiente, extraer los datos relevantes que se hayan seleccionado y clasificarlos para obtener la respuesta que indique el estado emocional del usuario.

Otro de los objetivos del proyecto es, cuando ya se haya generado todo el código, realizar las pruebas y validaciones hasta obtener el funcionamiento deseado y se detecten claramente los diferentes tipos de emociones.

Por último, si se hubiera adquirido el hardware, el último objetivo una vez se haya generado todo el código, habría sido lograr instalar el dispositivo e implementarlo en la plataforma de simulación vehicular.

Cabe destacar que el proyecto a realizar intenta asemejarse en mayor medida de lo posible a lo que podría ser un proyecto de aplicación real en la industria de la automoción. Tanto en tiempos de adquisición de señales como en desarrollar un software que tenga el menor coste computacional posible.

1.3. Alcance y beneficios

El alcance del proyecto se define en función de los focos de desarrollo de la siguiente manera:

- Análisis de los estudios basados en la detección de emociones basados en la electroencefalografía (EEG)



- Diseño de un sistema desarrollado en el entorno de Matlab capaz de adquirir y procesar las señales, extraer las características y clasificarlas en grupos.
- Testes y validaciones del sistema desarrollado para obtener resultados experimentales.
- Análisis de los protocolos de actuación frente a cada una de las posibles salidas del sistema.

Cada una de las fases está destinada a cumplir con su objetivo principal (por ejemplo, realizar una buena lectura de las señales) además de facilitar las siguientes tareas para lograr el objetivo final de conseguir una buena detección y clasificación de las respuestas emocionales del conductor.

El correcto desarrollo de un sistema capaz de analizar el estado emocional del conductor supondría un paso adelante en AIC frente a su línea de trabajo enfocada en la mejora del confort y la supervisión del conductor, por lo que se podría decir que el beneficio interno de la empresa es adquirir unas bases en la monitorización cerebral del conductor de cara a continuar en el futuro con el desarrollo del sistema.

Como beneficio personal, se pretende alcanzar unos conocimientos válidos en el área de trabajo de desarrollo de proyectos I+D (Investigación y desarrollo) como son llevar a cabo el desarrollo de un sistema de monitorización cerebral de usuario adaptado lo máximo posible a la realidad y su correcta programación y validación para el buen funcionamiento del mismo.

1.4. Resumen de contenido

- *Capítulo 1. INTRODUCCIÓN:* Se introducen los fundamentos del proyecto, desde la motivación de realizarlo hasta el alcance del mismo.
- *Capítulo 2. ESTADO DEL ARTE:* Se estudia el contexto científico en el que se sitúa el proyecto.
- *Capítulo 3. DEFINICIÓN DE LA SOLUCIÓN PROPUESTA:* En este capítulo se detalla el funcionamiento y diseño deseado y la selección de las diferentes características del sistema.
- *Capítulo 4. METODOLOGÍA:* Se detalla la metodología y los procedimientos llevados a cabo para lograr el correcto funcionamiento.
- *Capítulo 5. EJEMPLO DE APLICACIÓN:* Se expone un ejemplo de posible aplicación para el sistema presentado en el proyecto.
- *Capítulo 6. CONCLUSIONES Y TRABAJO FUTURO:* Se presentan las conclusiones extraídas a partir de la realización del proyecto y se expone el trabajo futuro propuesto.

Capítulo 2. ESTADO DEL ARTE

En este capítulo, se detalla el estado del arte actual sobre los temas influyentes en el proyecto. Para ello, es preciso realizar un análisis de los vehículos autónomos, las emociones y su impacto en el cerebro, el reconocimiento de emociones, los sistemas BCI y el confort vehicular para entender la finalidad del proyecto.

2.1. Vehículos autónomos

En los últimos años, los vehículos autónomos (VA) se han convertido en un tema muy popular, no sólo en el campo de la investigación sino también en el ámbito de las aplicaciones en la vida cotidiana [6].

Se espera que los VA aporten grandes ventajas y beneficios, causantes del auge actual de su mismo desarrollo. Entre ellas están el aumento de eficiencia en la conducción y reducción de las emisiones de CO₂, incremento de la seguridad vial, tanto para los conductores como para los peatones, y la reducción de accidentes.

Sin embargo, a pesar de haberse realizado muchos enfoques y prototipos alentadores, las estrategias de sistemas avanzados de conducción asistida (ADAS) existentes se centran demasiado en la "corrección" y, en cierta medida pasan por alto la personalidad humana y la inteligencia social [7].

Con el objetivo de reforzar la seguridad vial y, poniendo el foco en la personalidad humana, el proyecto se centrará en el desarrollo de un sistema brain-to-vehicle para el estudio de respuestas emocionales del conductor de un vehículo. De esta forma, implementado el sistema, se dotaría al vehículo de una autonomía de nivel 4 acorde a los niveles establecidos por el estándar SAE, donde el nivel 0 se corresponde con ninguna automatización de la conducción y el nivel 4 se corresponde con una automatización completa de la conducción, sin responsabilidad alguna del conductor, ante situaciones en las que el estado emocional del conductor pueda afectar a la conducción.

2.2. Las emociones y el cerebro

2.2.1 Qué son las emociones

Según Palmero y Fernández-Abascal (1999), el proceso emocional se desencadena por la percepción de condiciones internas y externas, que llegan a una primera evaluación. Como consecuencia de esta evaluación tiene lugar la activación emocional que se compone de una experiencia subjetiva, una expresión corporal o comunicación no verbal, una tendencia a la acción o afrontamiento y cambios fisiológicos que dan soporte a todas las actividades anteriores [8].

Así pues, se podría decir que una emoción es un estado afectivo que experimentamos, una reacción subjetiva al ambiente en el que nos encontramos, que, biológicamente tiene

como propósito incitar a la acción, nos da información sobre nuestro entorno y ayuda a comunicarnos con los demás.

2.2.2. Las emociones en el cerebro

El cerebro humano es el centro de todos los estímulos cognitivos y sensoriales que controlan las funciones corporales, produciendo complejas señales biopotenciales [10]. Por lo tanto, las emociones impactan en la actividad cerebral directamente, concretamente en el sistema límbico, una de las redes de neuronas más importantes en la aparición de estados de ánimo [9].

La teoría del sistema límbico sostiene que la percepción de lo que nos rodea, combinada con la información de los cambios corporales, genera experiencias afectivas, y esta integración de conocimiento ocurría en el sistema límbico [11].

Este está compuesto por estructuras del encéfalo conectadas entre sí, aunque el concepto de sistema límbico tiene más que ver con la función que tienen estas zonas que con su naturaleza como parte concreta y bien delimitada del cerebro. No obstante, se pueden diferenciar las principales partes del sistema límbico en: hipotálamo, hipocampo, amígdala y corteza orbitofrontal (Figura 2).

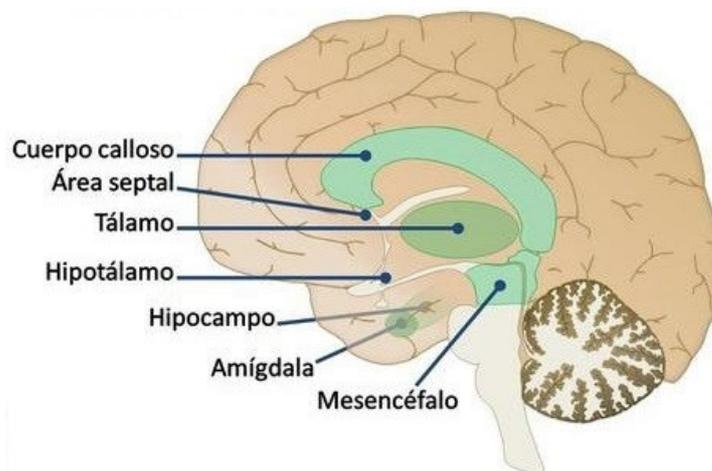


Figura 2. El sistema límbico [9]

El hipotálamo se encarga de liberar las hormonas necesarias para sentir emociones, mientras que el hipocampo controla los procesos mentales relacionados con la memoria y recuerdo de emociones.

La amígdala también forma parte de este proceso ya que está vinculada a la respuesta emocional que despiertan las situaciones experimentadas, siendo así el núcleo de control de las emociones en el cerebro.

Por otra parte, la corteza orbitofrontal, la cual envía ordenes emocionales hacia el lóbulo frontal, es la encargada de la planificación de nuestras acciones una vez recibimos un impulso irracional.

Para medir las señales de estas regiones del cerebro, en caso de utilizar señales eléctricas, se deben ubicar los electrodos en las zonas pertinentes. Basándonos en el Sistema Internacional de Posicionamiento de Electrodo 10-20, se han seleccionado como relevantes en este análisis los siguientes ocho electrodos: Fp1, Fp2, F3, F4, P3, P4, O1 y O2 (Figura 3).

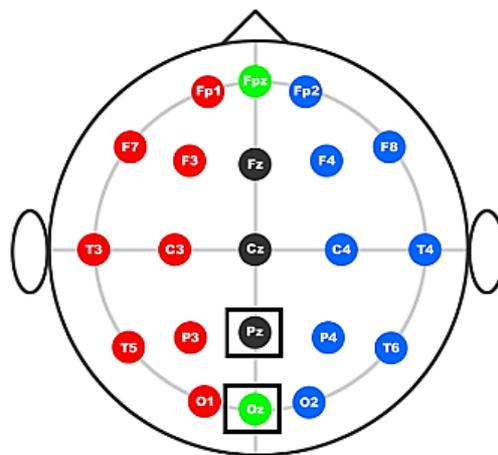


Figura 3. Sistema internacional de posicionamiento de electrodos 10-20 [12]

2.3. Reconocimiento de emociones

2.3.1. Neurociencia

Las emociones son parte de nuestra vida cotidiana, sin embargo, durante un tiempo quedaron fuera del mapa de la ciencia [13]. Estas eran consideradas un freno de la razón y muestra de animalidad y, por lo tanto, concluyeron en que había que controlarlas. Tras la primera y segunda guerras mundiales, asociaron las atrocidades a la razón humana, por lo que se reincorporaron en la explicación de las acciones humanas [14]. Ahora bien, es hasta los años setenta del siglo pasado que las emociones recobraron su importancia en la psicología no psicoanalítica y en la naciente neurociencia.

La neurociencia es el conjunto de ciencias que se encargan del estudio del sistema nervioso para comprender su funcionamiento con el fin de controlar emociones, conductas o funciones corporales básicas.

Estos estudios se realizan desde muchas perspectivas diferentes, entre sus áreas de investigación están la neurociencia molecular, clínica, cognitiva, neurofisiológica, etc.



Figura 4. Aplicación neuromarketing

La neurociencia cognitiva es la encargada de estudiar cómo el cerebro crea el pensamiento, la memoria, el lenguaje, etc. [15]. Esta tiene numerosas aplicaciones donde es preciso el estudio de las emociones, por ejemplo, en el mundo del marketing. Esto es denominado neuromarketing donde las empresas hacen uso de esta disciplina para inducir a los usuarios a consumir más un producto.

2.3.2. Tipos de interfaces cerebro-computadora

A lo largo de los años se han ido desarrollando diversos sistemas capaces de captar el estado cerebral, entre ellos, los más comunes son los siguientes:

- Electroencefalograma (EEG): Se analiza y registra la actividad eléctrica del cerebro con sensores en el cuero cabelludo, detecta los cambios en las corrientes eléctricas en forma de ondas cerebrales.
- Resonancia magnética funcional (fMRI): Mide y mapea la actividad cerebral a través de la detección de cambios en el flujo sanguíneo. Permite explorar estructuras pequeñas y profundas del cerebro.
- Magnetoencefalograma (MEG): Detecta los cambios en los campos magnéticos inducidos por la actividad eléctrica del cerebro cuando el usuario se expone a un estímulo.
- Tomografía por emisión de positrones (PET): Detecta y analiza la distribución tridimensional que adopta un radiofármaco de vida corta que es inyectado por vía intravenosa. Se detectan cambios en la composición química o flujo de fluidos en estructuras pequeñas y profundas del cerebro.

- Steady state topography (SST): Se mide la variación en potenciales evocados visuales medidos en la actividad del sujeto cuando se expone a estímulos visuales.
- Espectroscopia funcional del infrarrojo cercano (fNIR): Mide la absorción de la luz NIR en la hemoglobina de la sangre con o sin oxígeno y proporciona información sobre la actividad funcional cerebral similar a los estudios de resonancia magnética funcional.

2.4. Brain Computer Interface (BCI)

Las emociones son procesos psicofisiológicos desencadenados por la percepción consciente y/o inconsciente de un objeto o situación y suele asociarse con el estado de ánimo, el temperamento, la personalidad y la disposición, y la motivación.

Las emociones desempeñan un papel importante en la comunicación humana y pueden expresarse verbalmente a través de un vocabulario emocional, o mediante la expresión de señales no verbales como la entonación de la voz, las expresiones faciales y los gestos.

La mayoría de los sistemas contemporáneos de interacción persona-ordenador son deficientes a la hora de interpretar esta información, es decir, son incapaces de identificar los estados emocionales humanos y utilizar esta información para decidir las acciones adecuadas a ejecutar [16].

El interés y auge creciente de nuevas soluciones aplicadas a los VA se debe al avance exponencial del desarrollo de las tecnologías involucradas en este entorno [6]. Gracias a estos avances, se han desarrollado tecnologías como el Brain-Computer Interface (BCI), un sistema que adquiere señales cerebrales, las analiza y las traduce en ordenes que se transmiten a un dispositivo de salida para llevar a cabo la acción deseada [17].

El objetivo de la computación afectiva es llenar este vacío detectando emociones que se producen durante la interacción persona-ordenador y sintetizar las respuestas emocionales [16].

Las señales más estudiadas y también las que se van a utilizar en el proyecto, son las señales eléctricas medidas a partir de electrodos.

Un sistema BCI consta de 4 componentes secuenciales:

1. Adquisición de señales
2. Extracción de características
3. Traducción de características
4. Salida del dispositivo

En la actualidad, la mayoría de logros del desarrollo de sistemas BCI se limita casi exclusivamente a la investigación. Sin embargo, las interfaces cerebro-ordenador podrían llegar a utilizarse de forma rutinaria para numerosas aplicaciones, entre ellas para aumentar los resultados motores naturales de conductores [17].

Así pues, el proyecto se basa en el desarrollo de un sistema BCI aplicado a la conducción, siguiendo las componentes mencionadas anteriormente de adquisición de señales y procesado de las mismas, extracción de características y entrenamiento y validación para que el clasificador seleccionado proporcione la salida del dispositivo.

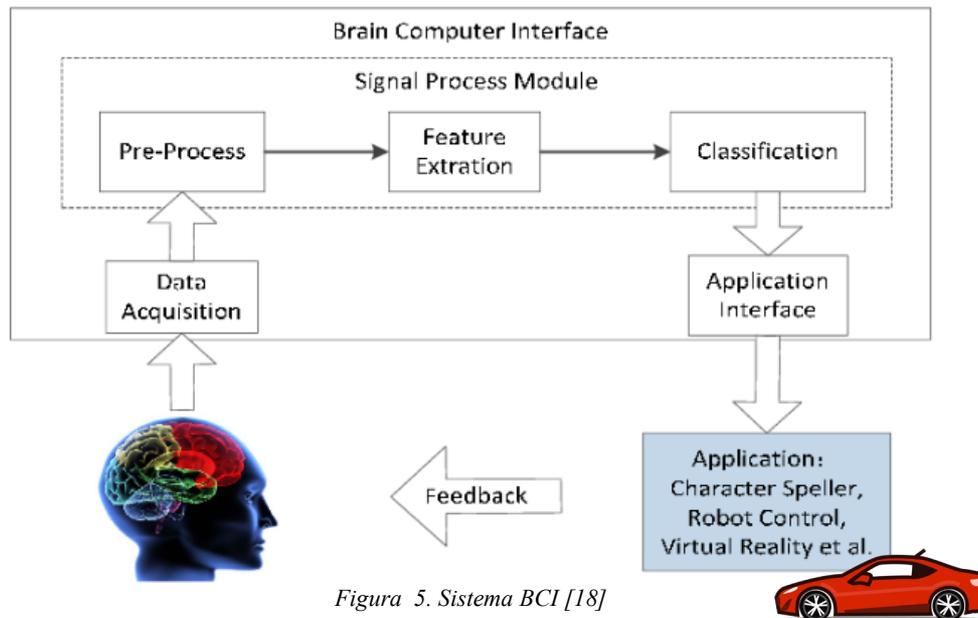


Figura 5. Sistema BCI [18]

2.5. Confort vehicular

El confort en la conducción es un factor importante puesto que pasamos una gran cantidad de tiempo en nuestra vida al volante, por lo que conducir de una forma cómoda resulta un componente a tener en cuenta en el desarrollo de nuevos vehículos. Para poder controlar este confort se debe comprender cuál es la razón por la que lo perdemos. La razón del malestar en la conducción (por ejemplo, mareos) se debe a la excitación de la frecuencia natural del ser humano causada por las vibraciones.

Cuando se habla de frecuencia natural nos referimos a las frecuencias a las que una estructura vibraría si se desvía y se suelta [19]. En cierto modo, el cuerpo se modela como si se tratase de una sola masa, de esta forma emula el amortiguamiento al que está sometido.

Las frecuencias naturales del ser humano se encuentran principalmente entre 1 y 200 Hz, siendo las frecuencias de entre 1-10 Hz las más molestas para los seres humanos. En la Figura 6 se aprecia el rango frecuencial de las diferentes formas de excitación e interacción con el cuerpo humano.

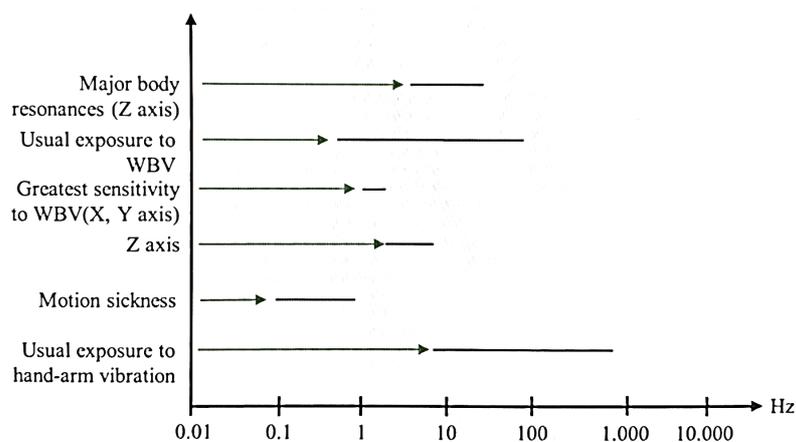


Figura 6. Rango frecuencial de los efectos de las vibraciones en seres humanos

Así pues, para conseguir el confort se debe evitar situar al vehículo en frecuencias dentro del rango de las frecuencias naturales del ser humano. Esto se puede controlar a través de la suspensión del vehículo, que da opción a moldear las variables relevantes en la mejora del confort en la conducción: magnitudes de aceleración, tiempo y frecuencia.

Las magnitudes de aceleración generalmente no deben ser elevadas ya que generan daños físicos en el cuerpo humano como podría ser, por ejemplo, un latigazo cervical por una aceleración muy brusca. Además, las frecuencias deben estar fuera del rango de las frecuencias naturales del ser humano (1-10 Hz), incluso llegando a frecuencias no perceptibles por este. Todo esto lleva una variable temporal, de forma que tanto los valores de magnitud de la aceleración como las frecuencias próximas a las naturales no deben prolongarse demasiado en el tiempo para lograr el confort vehicular.

Como se ha comentado, estos factores pueden variar modificando la suspensión del vehículo. Existen dos tipos de suspensiones, activa y pasiva. La suspensión pasiva se caracteriza por tener parámetros constantes y, por lo tanto, no necesitar ningún actuador o sensor.

Sin embargo, la suspensión activa es un tipo de sistema capaz de generar la fuerza y suavidad cuando se requiera. Uno de sus principales objetivos es aportar confortabilidad a los sistemas vehiculares. Esta funciona a través de un sistema electrónico que controla diferentes componentes hidráulicos, entre ellos bombas, actuadores y servoválvulas. Estos se encargan de regular el aire que extrae la válvula en los estados de tracción y compresión del conjunto de muelle y amortiguador.



Capítulo 3. DEFINICIÓN DE LA SOLUCIÓN PROPUESTA

En este capítulo se va a describir tanto el diseño propuesto como el funcionamiento deseado del sistema, exponiendo todas las opciones posibles para cada uno de los factores determinantes y concluyendo con la justificación de la selección escogida en cada caso.

3.1. Funcionamiento

El objetivo es desarrollar un sistema capaz de clasificar el estado emocional del usuario a través de la lectura de señales cerebrales y actuar en consecuencia a este. Para esto, es necesario separar el proyecto en bloques. El primer bloque sería la adquisición de las señales cerebrales a través de un dispositivo hardware para introducirlas en el software.

El siguiente sería el encargado de procesar la señal para adecuarla a la correcta extracción de datos. Esta extracción se realiza en el siguiente bloque, donde se extraen datos en ventanas de un tiempo determinado, en el que se capturan los valores de las características determinantes en el análisis. Cuando se hayan obtenido los datos, el siguiente paso sería filtrarlos con un clasificador para obtener como resultado el estado emocional del conductor.

El último bloque se encarga de, una vez se disponga de la información sobre el tipo de emoción, activar un protocolo de actuación pertinente. En este caso los protocolos que se han planteado se basan en mejorar el confort del conductor seleccionando el modo de conducción que más se adapte a sus necesidades en función de su estado emocional. Generalmente, se distinguen cuatro modos de conducción:

-Modo sport/dinámico: se aprovecha al máximo la potencia del motor, ofreciendo una conducción más activa, divertida, y rápida.

-Modo confort/normal: modo por defecto del vehículo.

-Modo eco: limita la velocidad máxima, suaviza la dirección, disminuye ligeramente la potencia, y economiza el aire acondicionado, procurando que todo el sistema sea más eficiente para que se genere el menor gasto de combustible posible.

-Modo individual: permite que las diferentes funciones y sistemas del vehículo sean configurados a preferencia y necesidad del conductor.

3.2. Diseño general

Una vez descritos los bloques necesarios para el funcionamiento deseado, se va a ilustrar por medio de un diagrama de bloques la estructura del sistema (Figura 7).

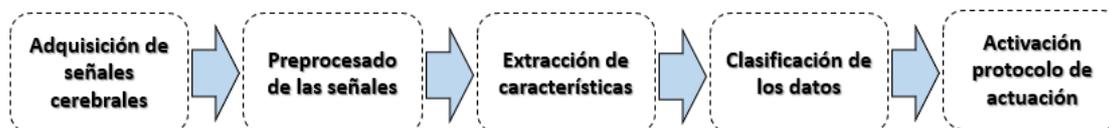


Figura 7. Diseño general del sistema

Así pues, el diseño presentado es el que debería seguir el sistema de monitorización cerebral a desarrollar.

3.3. Especificaciones del sistema

En este apartado se van a presentar las opciones posibles para cada uno de los factores relevantes del sistema desde las características a extraer hasta los dispositivos físicos existentes. Además, se va a realizar la selección de cada uno de ellos con su justificación.

3.3.1. Tipo de interfaz cerebro-computadora

Los tipos de métodos de medición de señales cerebrales se han presentado en el apartado 2.3.2. de la memoria. Tras analizar cada uno de los tipos, se ha realizado una tabla comparativa de las características relevantes que necesita el interfaz que se seleccione (Figura 8).

	PORTÁTIL	RESOLUCIÓN TEMPORAL	RESOLUCIÓN ESPACIAL	MÉTODO	COSTE
EEG	Sí	Alta	Baja	No invasivo	Bajo
fMRI	No	Baja	Alta	No invasivo	Alto
MEG	No	Alta	Alta	No invasivo	Muy altos
PET	No	Baja	Alta	Invasivo	Alto
fNIR	Sí	Baja	Baja	No invasivo	Alto

Figura 8. Tabla comparativa interfaces cerebro-computadora

Finalmente se ha decidido escoger el electroencefalograma (EEG) como método para la adquisición de las señales cerebrales ya que es el más adecuado para este tipo de aplicación. Entre las modalidades no invasivas de monitorización de la actividad cerebral en la vida cotidiana, el EEG es el único que puede utilizar sensores secos y puede llevarse durante la locomoción libre [20].

El comportamiento dinámico del cerebro se ve claramente reflejado en el registro de las señales del electroencefalograma, por ello es un método muy utilizado para medir la actividad neurofisiológica del cerebro en forma de señales eléctricas [21].

Este permite la detección de anomalías en la actividad eléctrica cerebral típicas de alteraciones que afectan al sistema nervioso cerebral [22].

El análisis de estas señales puede realizarse en vivo o en diferido. Aunque lo más frecuente es utilizar estos sistemas en diferido, el electroencefalograma también puede utilizarse en vivo para detectar monitorizar el nivel de asistencia de las personas.

Este tipo de interfaz se usa en numerosos artículos de investigación sobre detección de emociones, por ejemplo, en [23] se construye un sistema de detección afectiva del paciente con una tasa de acierto del 65 %.

Los impulsos eléctricos procedentes del cerebro que se registran en señales EEG pueden corresponder a diferentes ritmos cerebrales, también conocidos como bandas cerebrales. Existen cinco tipos de bandas cerebrales distinguidas por sus rangos de frecuencia [24]:

- *Banda delta (0.5 - 4 Hz)*: estas ondas son las de menor frecuencia y se asocian a estados de sueño profundo. Además, al ser tan lentas es muy fácil confundirlas con ruido.
- *Banda theta (4 - 8 Hz)*: estas tienen una amplitud grande y se asocian a estados de meditación y creatividad y a procesos en los que interviene la memoria [25].
- *Banda alpha (8 - 12 Hz)*: estas ondas están más relacionadas con estados de relajación o calma, donde la misma ausencia de estas también indica concentración o ansiedad y son las más comunes en la investigación.
- *Banda beta (12 - 30 Hz)*: estas ondas se asocian a un estado normal y activo, también se asocian a estados de actividad mental intensa, concentración, agitación y ansiedad.
- *Banda gamma (30 - 45 Hz)*: por último, todavía no se ha encontrado una clara relación con ningún estado de estas ondas, pero estudios como [26] sugieren que podrían estar implicadas en el proceso de percepción consciente.

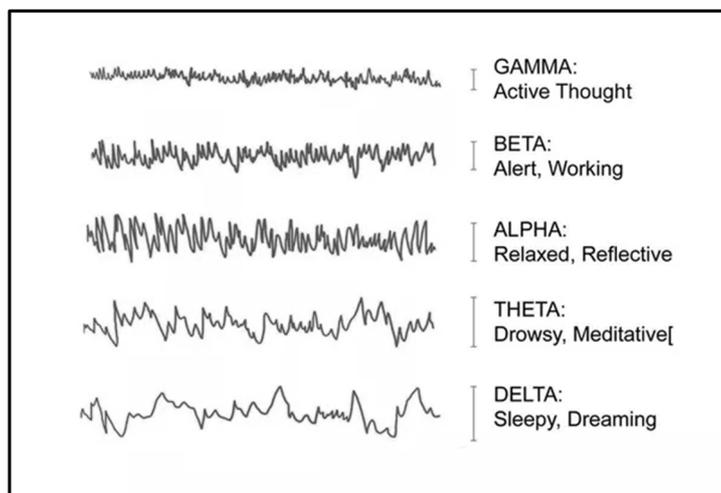


Figura 9. Tipos de ondas cerebrales [32]

3.3.2. Características a extraer de las señales EEG

Para poder extraer información de las señales EEG, es preciso extraer cierta información de ellas. Existen numerosas características y métodos de extracción para el reconocimiento de emociones a partir de señales cerebrales, sin embargo, dentro de estas, algunas técnicas se realizan en el dominio del tiempo, frecuencia, tiempo-frecuencia, etc.

En el dominio del tiempo suelen utilizarse características como el valor medio y la potencia [27]. En el dominio de la frecuencia, Wang et al. extrajeron características para la clasificación como la media de la señal, la desviación estándar de la señal o las medias de la primera y segunda diferencia de las señales normalizadas [28]. Con respecto al análisis de tiempo-frecuencia se basa en el espectro de las señales de EEG, con características como la energía, la potencia, la densidad espectral de potencia (PSD) y la entropía diferencial de las bandas cerebrales [29]. También se utilizan otras características extraídas de la combinación de electrodos, como la asimetría de los electrodos en diferentes regiones del cerebro [30].

Huang et al. propusieron el uso de la transformada EMD en 1998 [31], que, a diferencia de la DWT, puede descomponer las señales en funciones intrínsecas (IMF) automáticamente, estas representan diferentes componentes de frecuencia de las señales originales. Al aplicar la transformada de Hilbert a las IMF, podemos obtener información de fase instantánea de ellas. Por lo tanto, EMD es adecuado para el análisis de secuencias no lineales y no estacionarias, como las señales neuronales.

Por ello, se va a escoger como método de extracción de características para el proyecto. La razón principal por la que se ha decidido seleccionar este método es debido a que trabaja en varios dominios (dominio del tiempo, dominio de frecuencia y energía) la precisión puede aumentar. En este artículo se trabaja con funciones intrínsecas, donde siguiendo con el artículo [1] las tres características extraídas son las siguientes:

1. Primera diferencia de fase de las funciones
2. Primera diferencia de series de tiempo de las funciones
3. Energía normalizada de las funciones

Para calcular las funciones intrínsecas que se ha mencionado, se hará uso de la transformada EMD o descomposición empírica en modos.

3.3.2.1. Transformada EMD (Descomposición Empírica en Modos)

La transformada EMD, descompone las señales en funciones intrínsecas, donde estas representan diferentes componentes de frecuencia de las señales originales y deben satisfacer dos condiciones: el número de puntos extremos y cruces de cero deben ser iguales o diferir en uno como máximo; y en cada punto, el valor medio a partir de la

envolvente superior e inferior debe ser cero. Así pues, una señal de entrada $x(t)$ puede ser expresada como:

$$x(t) = \sum_{n=1}^L (imf_n) + r \quad (1)$$

Siendo L igual al número máximo de funciones intrínsecas (IMF) que queremos calcular menos 1 y r el resto de la función que se corresponde con la entrada sin su procesamiento posterior.

Por lo que esta es una combinación de los componentes de la función intrínseca y el resto, y, se podría decir, que la transformada EMD funciona como un filtro de paso alto adaptativo (Figura 10).

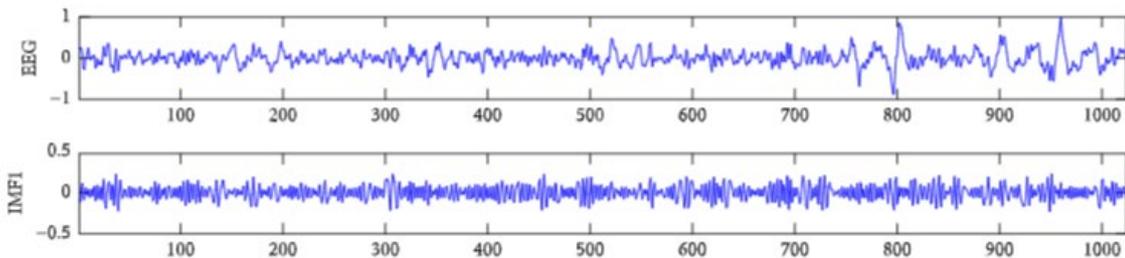


Figura 10. Función intrínseca a partir de señal EEG

3.3.2.2. Primera diferencia de fase

La primera diferencia de fase representa la intensidad del cambio de la señal en el dominio de la frecuencia. Cuando se dispone de las funciones, se obtendrá la señal analítica realizando la transformada de Hilbert $z(n)$ de cada una de ellas.

$$z(n) = x(n) + jy(n) \quad \text{o} \quad z(n) = A(n)e^{j\varphi(n)} \quad (2)$$

Donde el módulo de $A(n) = \sqrt{x(n)^2 + y(n)^2}$, y, por tanto, $\varphi(n) = \arctan\left(\frac{y(n)}{x(n)}\right)$ es la fase instantánea.

De esta manera, la primera diferencia de fase queda definida por la siguiente ecuación, que mide la intensidad de cambio de fase y representa el significado físico de la frecuencia instantánea:

$$D_p = \frac{1}{(N-1)} \sum_{n=1}^{N-1} |\varphi(n+1) - \varphi(n)| \quad (3)$$

3.3.2.3. Primera diferencia de series de tiempo

La primera diferencia de series de tiempo representa la intensidad del cambio en la fase de la señal en el dominio del tiempo. En investigaciones como [27] han sacado la conclusión de que la variación de las series temporales del EEG puede reflejar diferentes estados emocionales. Para una función IMF con N puntos, se puede definir D_t como:

$$D_t = \sum_{n=1}^{N-1} |imf(n+1) - imf(n)| \quad (4)$$

3.3.2.4. Energía normalizada

Por último, la energía normalizada describe el peso del componente de oscilación actual y está definida como:

$$E_{norm} = \frac{\sum_{n=1}^N imf^2(n)}{\sum_{n=1}^N s^2(n)} \quad (5)$$

Siendo $s(n)$ la señal original EEG, por lo que el numerador es la energía de la función IMF y el denominador representa la energía original de la EEG.

3.3.4. Modelos de clasificación emocional

Generalmente, las emociones han sido definidas dimensionalmente en la investigación. Primeramente, Descartes planteó la existencia de seis emociones primarias: asombro, amor, odio, deseo, alegría y tristeza. El filósofo Wilhelm Max Wundt, sugirió que las emociones se dividen en tres dimensiones basadas en contrarios: agradable-displacentero, excitar-someter y tensión-relajación. Más adelante, Harold Schlosberg definió tres similares dimensiones de las emociones: agrado-desagrado, atención-rechazo y nivel de activación.

Estos modelos tienen como fin ubicar las emociones en un plano bi o tridimensional y normalmente se basan en las dimensiones de valencia y excitación. Se han desarrollado numerosos modelos dimensionales de las emociones, dentro de estos los más comunes en la actualidad son los siguientes:

1. Modelo circunflejo (Russell)

Este modelo sugiere que las emociones se distribuyen en un espacio circular bidimensional, con dimensiones de excitación y valencia. La excitación se representa en el eje vertical y la valencia en el horizontal (Figura 11). Estos modelos se usan comúnmente para analizar expresiones faciales emocionales o estados afectivos.

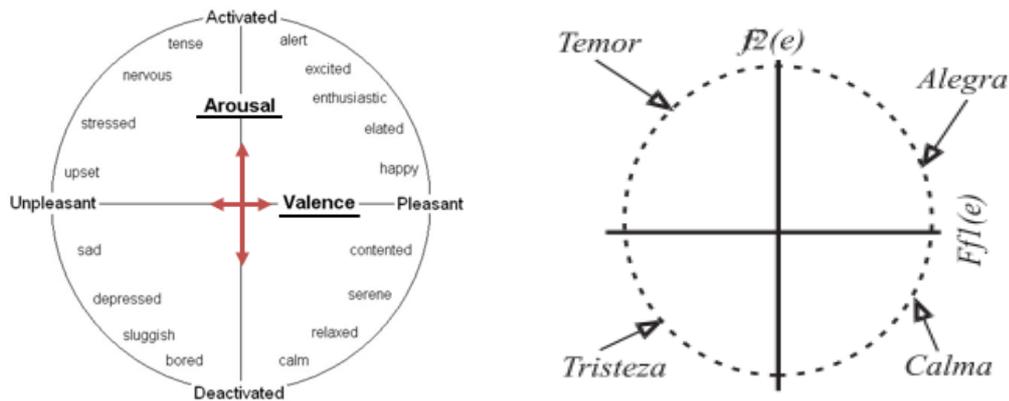


Figura 11. Modelo de Russell

2. Modelo vectorial

Este es un modelo bidimensional que consta de vectores que apuntan en dos direcciones, asume que, siempre hay una dimensión de excitación subyacente y que la valencia determina la dirección en que se encuentra una emoción. Estos modelos se usan normalmente en la prueba de estímulos de palabras e imágenes.

3. Modelo de Plutchik

Al igual que el modelo de Russell, sugiere una representación circunfleja, donde las emociones se agrupan en ocho categorías primarias: éxtasis, admiración, temor, asombro, odio, furia y pena (Figura 12).

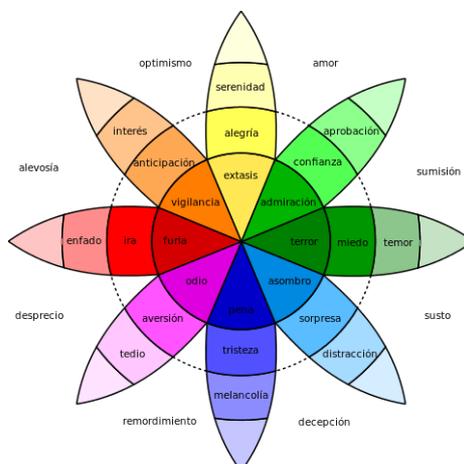


Figura 12. Modelo de Plutchik

4. *Modelo consensual (PANA)*

Este modelo llamado Activación Positiva - Activación Negativa sugiere que el afecto positivo y el afecto negativo son dos sistemas separados. En el eje vertical representa un afecto positivo de bajo a alto y el eje horizontal representa un afecto negativo de bajo a alto. Las dimensiones de valencia y excitación se encuentran en una rotación de 45 grados sobre estos ejes.

Finalmente, dado que es el modelo que siguen la mayor parte de las investigaciones que se han encontrado en el contexto científico, se ha decidido optar por el modelo de Russell en la clasificación de emociones, donde los ejes representan la valencia (eje x) y la excitación (eje y) del usuario, dividiendo así las emociones en cuatro cuadrantes relativos a los estados emocionales de tristeza, alegría, preocupación/temor y calma (Figura 11).

3.3.5. Clasificadores

Existen muchos métodos para obtener los resultados de esta clasificación, por ejemplo: LDA (Linear Discriminant Analysis), SVM (Support Vector Machine), GBDT (Gradient-Boosted decision trees), RSRM, Redes neuronales...

Aunque cada uno de ellos engloba unas características diferentes, es complicado llegar a saber cuál es el más adecuado en esta situación, ya que depende de los datos de los que se disponga.

Para poder realizar un análisis amplio en el espectro de clasificadores de los que disponemos, se va utilizar la herramienta de MATLAB Classification Learner. Esta app permite entrenar modelos para clasificar datos y explorar el proceso de machine learning supervisado mediante estos clasificadores.

Además, permite explorar los datos, seleccionar características, especificar esquemas de validación, entrenar modelos y evaluar los resultados. Esta app incluye clasificadores como árboles de decisión, análisis discriminantes (LDA), máquinas de vectores de apoyo (SVM), regresión logística, vecinos más próximos Naive Bayes, aproximación de kernel, ensemble y clasificación de redes neuronales (KNN) [33].

Uno de los métodos más utilizados en el aprendizaje supervisado en machine learning es el basado en árboles de decisión ya que tienen una alta precisión, estabilidad y facilidad de interpretación. Además, al no ser modelos lineales, se adaptan a problemas tanto de clasificación como de regresión.

El análisis discriminante es un método utilizado en estadística, reconocimiento de patrones y aprendizaje en busca de una combinación lineal de rasgos que caracterizan o separan dos o más clases de objetos o eventos [34].

En cuanto al clasificador tipo SVM, son una técnica de machine learning que encuentra la mejor separación posible entre clases. Este tipo de clasificador es eficaz en espacios de grandes dimensiones, incluso cuando el número de dimensiones es mayor al número de muestras.

Las redes neuronales permiten extraer información útil y producir inferencias a partir de los datos disponibles gracias a su capacidad de aprendizaje. Funcionan como reconocedores de patrones altamente tolerantes a errores por lo que permiten combinar las cualidades del razonamiento humano con la lógica precisa y la memoria de los ordenadores. Además, las redes neuronales son capaces de funcionar con datos incompletos, imprecisos o con gran cantidad de ruido. Se autoajustan a medida que se entrenan con la información disponible, de forma que aprenden a reconocer paulatinamente todos los casos del conjunto de datos utilizado para su entrenamiento [35].

Todos estos métodos podrían resultar útiles en el análisis propuesto, por lo que se realizarán las pruebas con cada uno de ellos para poder seleccionar el más adecuado, es decir, el que tenga más precisión.

3.3.6. Base de datos

Para poder entrenar al clasificador como se ha mencionado en el apartado anterior, se necesitan datos de entrenamiento. Para esto se va a hacer uso de la base de datos llamada DEAP dataset. Esta es una base de datos pública para estudiantes e investigadores orientada al análisis de los estados afectivos del ser humano.

En esta se registraron los electroencefalogramas en diferentes electrodos y otras señales fisiológicas para obtener el estado emocional de 32 participantes durante la emisión de 40 fragmentos de un minuto de videos musicales. Los mismos participantes evaluaron los videos en una escala del 1-9 en de los niveles de excitación, valencia, agrado/desagrado, dominancia y familiaridad.



Figura 13. Base de datos DEAP dataset [41]

Esta base de datos proporciona varios ficheros relativos a la extracción de datos, desde la evaluación de los participantes que se ha mencionado hasta las señales cerebrales extraídas en los experimentos y los niveles de valencia, excitación, dominancia y familiaridad extraídos de ellas.

En este caso, vamos a utilizar el fichero de datos *data_preprocessed_matlab.zip*, que contiene 32 ficheros *.mat* relativos a las señales preprocesadas (filtradas entre 4-45 Hz, con una frecuencia de muestreo de 128 Hz) de cada participante de los diferentes electrodos. Cada uno de estos ficheros contiene dos arrays: *data* (video x canal x dato) y *labels* (video x etiqueta). Con el primero se puede extraer la señal cerebral de un electrodo y durante la emisión de un video en concreto, y con el segundo se puede visualizar el valor de valencia, excitación, dominancia o agrado entre 1 y 9 obtenido en la extracción de los datos en un video concreto.

Array name	Array shape	Array contents
data	40 x 40 x 8064	video/trial x channel x data
labels	40 x 4	video/trial x label (valence, arousal, dominance, liking)

Figura 14. Contenido fichero *data_preprocessed_matlab.zip*

Estas señales relativas a 32 electrodos diferentes, están muestreadas a 128 Hz y filtradas con un filtro de paso bajo de 4 – 45 Hz. Cada señal tiene 63 segundos, los 3 primeros que no contienen información ya que son previos al video, y los 60 siguientes que son los de la señal válida.

3.3.7. Protocolos de actuación

Cada una de las emociones que sentimos puede suponer una reacción diferente al volante. Por ejemplo, estar alegre se puede asociar con una conducción más rápida y fluida, sentirse triste se relaciona más con reacciones lentas, sentirse estresado o preocupado refleja en una conducción más agresiva y distraída [33].

Puesto que el sistema va a clasificar las emociones según el modelo Russell, las salidas que se van a obtener son cuatro: tristeza, alegría, preocupación/temor y calma. Para cada uno de los estados emocionales, se debe configurar o asignar un modo de conducción.

En primer lugar, para el estado emocional de alegría o felicidad, se activará el modo sport. En este modo la dirección se endurece y se hace más directa, además, aumenta la respuesta del acelerador, e incrementa la rigidez de la suspensión. Así pues, la aceleración sería más brusca y rápida, con magnitudes de hasta 0.9 m/s², pero sin llegar a entrar en frecuencias naturales que supongan la pérdida del confort. Normalmente, en este tipo de conducción los sistemas de seguridad funcionan de manera más permisiva ya que empiezan a operar bajo un margen de control menor.

Cuando el conductor se encuentre relajado o calmado, el modo que se activará será el modo confort, es decir el predeterminado del vehículo, donde los sistemas se encuentran equilibrados y balanceados.

Como se ha comentado anteriormente, el confort está directamente relacionado con la suspensión del vehículo. De esta forma, los modos de conducción para estados de preocupación o tristeza se van a personalizar en función de las necesidades del conductor en cada caso.

El modo de conducción personalizado para tristeza sería uno con aceleración más lenta, por ejemplo, de una magnitud entorno a unos 0.3 m/s². En cambio, el modo de conducción que se va a establecer para el estado de preocupación se acercaría más a un modo sport elevando la magnitud de aceleración, pero al contrario de este, con menos permisividad en los sistemas de seguridad puesto que el conductor puede estar más distraído.

3.3.8. Protocolos de comunicación con la plataforma de simulación

Normalmente, para la conexión de sistemas auxiliares de monitorización de usuario al simulador de conducción de AIC, se utilizan comunicaciones a través de ethernet, en concreto, modelos TCP/IP. Dentro de estos, generalmente se usan los protocolos UDP y TCP.

El protocolo TCP es el más común ya que es un protocolo orientado a la conexión, es decir, que además de transmitir la información entre dispositivos, también verifica la recepción de esta. De esta manera, este protocolo tiene mayor fiabilidad.

El protocolo UDP funciona de manera similar, sin embargo, este no verifica la recepción de los datos recibidos por el otro dispositivo. Esto, aunque tiene peor fiabilidad, resulta en un protocolo con mayor velocidad de transferencia. Por esta razón, sus usos son más comunes en aplicaciones como transmisión de voz o streaming, donde es más relevante la velocidad que una pérdida de datos puntual.

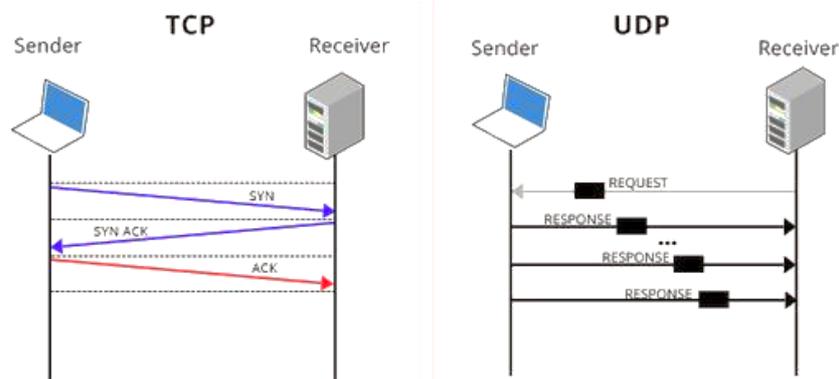


Figura 15. Protocolos de comunicación con el simulador



Basándose en las necesidades del sistema propuesto, se ha escogido comunicarse con el simulador por protocolo UDP ya que se trata de un sistema donde la pérdida de datos en un momento puntual no resultaría un problema. Esto se debe a que, aplicando la lógica, el usuario no cambia de estado emocional en cuestión de milisegundos, por lo que resulta más relevante la velocidad de transmisión que la fiabilidad del sistema.

3.3.9. Software

En cuanto al software que se ha empleado, en una primera instancia dependería del hardware que se va a utilizar. Sin embargo, todas las opciones de dispositivo planteadas para usar como dispositivo de medida EEG son compatibles con MATLAB/Simulink por lo que se va a optar por el uso de este software. Además, el simulador vehicular de la empresa trabaja en este mismo software por lo que es la mejor opción para compatibilizar el sistema con este.

Las señales de entrada se introducirán en el programa ya que este software permite crear un control rápido con sistemas físicos en MATLAB y Simulink.

Estas señales pasarán por un filtrado para poder introducirse adecuadamente en la función de extracción de las características. Además, se programará el clasificador escogido desde MATLAB, para esto, se cargarán los datos y ajustarán los tamaños de la base de datos de entrenamiento y se entrenará hasta obtener un clasificador con buenos resultados.

Cuando el clasificador esté entrenado, este se usará para diferenciar las emociones seleccionadas y establecer los protocolos de actuación frente a cada una de ellas.



Capítulo 4. METODOLOGÍA

En este capítulo se va a desarrollar el proceso que ha sido necesario para llevar a cabo el proyecto desde su base, hasta lograr el funcionamiento expuesto.

Para esto ha sido necesario pasar por diferentes fases, ejecutando rigurosamente las tareas requeridas para alcanzar los objetivos a corto plazo de cada una de ellas.

Antes de comenzar con la parte práctica del proyecto, ha sido necesaria la definición de estas tareas y su duración en el tiempo del que disponemos, esto se ilustra en un cronograma en el subapartado 4.1.

En primer lugar, ya que se ha partido desde cero, ha sido necesario un riguroso estudio bibliográfico para contextualizar la naturaleza del proyecto.

Siguiendo con el desarrollo del proyecto, el componente fundamental para que este funcione correctamente es el código, por lo que se podría decir que este es el proceso más largo e importante. Este se va a detallar en el subapartado 4.4. dónde se exponen todas las tareas que ha sido necesario realizar. Cuando se haya desarrollado el código el siguiente paso es realizar las pruebas y análisis de los datos obtenidos (4.5.) para analizar las características escogidas.

Cuando estas se hayan seleccionado, se han desarrollado los protocolos de actuación pertinentes adecuándose a cada tipo de situación en función del estado emocional que se detecte (4.6.).

Una vez se ha desarrollado todo el software del proyecto, procedemos a realizar los pertinentes test y validaciones para concluir con su correcto funcionamiento, esto se expone en el subapartado 4.7., donde se presentan los resultados obtenidos.

En el caso de adquirir el hardware propuesto, el siguiente paso sería la implementación del mismo al simulador para trasladar estas pruebas a la realidad, de forma que se aproxime al objetivo en la mayor medida de lo posible (subapartado 4.8.).

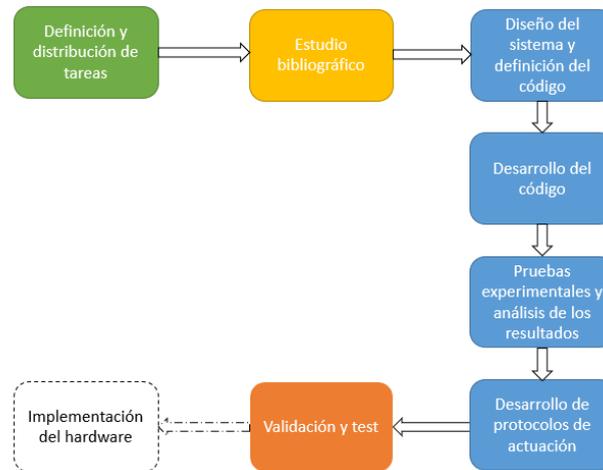


Figura 16. Esquema metodología

4.1. Definición y distribución de tareas

Con el fin de distribuir correctamente las tareas que requiere el proyecto, se ha realizado un cronograma. Esto se representa con un diagrama de Gantt donde las tareas se distribuyen en el tiempo con una serie de objetivos a corto plazo para llevar al final del proyecto en un tiempo establecido (Figura 21).

Desarrollo e integración de sistemas brain-to-vehicle para el estudio de respuestas emocionales en un simulador de carretera	MARZO		ABRIL		MAYO		JUNIO	
	15-31 MARZO	01-15 ABRIL	15-30 ABRIL	01-15 MAYO	15-31 MAYO	1-15 JUNIO	15-30 JUNIO	
META DEL PROYECTO								
FASE 1								
Estudio bibliográfico								
Selección del dispositivo								
Propuesta estructural y características del sistema								
Análisis de requerimientos de conectividad								
FASE 2								
Definición del software								
Desarrollo del software								
Pruebas experimentales								
Validación y test								
FASE 3								
Definición final de variables a integrar en el sistema								
Establecimiento de protocolos de actuación								
Desarrollo de ejemplo básico de aplicación								

Figura 17. Diagrama de Gantt

De esta manera, el desarrollo del proyecto queda dividido en cuatro fases en el tiempo con diferentes tareas a realizar.

Fase 1:

- Estudio de las referencias bibliográficas en las áreas de monitorización cerebral para el estudio de respuestas emocionales y, en particular, de su aplicación al ámbito de la movilidad autónoma y el confort de marcha.
- Selección de dispositivo de adquisición de señales eléctricas de actividad cerebral.
- Propuesta de estructura y características del sistema de adquisición de señales y control vehicular basado en la respuesta emocional.
- Análisis de requerimientos de conectividad e integración hardware/software con el simulador de conducción del VDC.

Fase 2:

- Definición de funciones software brain-to-vehicle específicas a integrar en el sistema.
- Desarrollo de funciones del sistema utilizando como programa Matlab.
- Pruebas experimentales del sistema.
- Validación y test.

Fase 3:

- Definición del conjunto de variables significativas a integrar en el sistema de monitorización de la respuesta emocional.
- Establecimiento de protocolos de actuación en base a las variables monitorizadas.
- Desarrollo de ejemplo básico de aplicación.

4.2. Estudio bibliográfico

Ha sido necesario, primeramente, el estudio de las emociones en el cerebro, los métodos existentes para medir estas señales cerebrales y los dispositivos disponibles para llevarlo a cabo. Además, ha sido necesario el estudio de los sistemas BCI ya que el sistema se basa en este modelo, que tiene como pasos desde la adquisición de la señal cerebral y la extracción de sus características hasta la clasificación de la misma. Asimismo, ha sido necesario el estudio de los modelos de clasificación de emociones, así como la forma de llevar a cabo esta clasificación en nuestro proyecto. También, al tratarse de un sistema orientado a la conducción, se ha estudiado el confort vehicular para poder desarrollar los protocolos de actuación. Este contenido se recoge detalladamente en el estado del arte (Capítulo 2) y la definición del sistema propuesto (Capítulo 3).

4.3. Definición del código

En este apartado se va a explicar las funciones que han sido necesarias desarrollar para lograr el funcionamiento requerido del sistema. Para definir las tareas que debe realizar, lo primero a realizar es identificarlas. En primer lugar, se deberá tener en cuenta que al obtenerse los datos de las señales EEG de una base de datos (DEAP dataset), estas se encuentran muestreadas a una frecuencia y un rango de frecuencias específicos, por lo que deberemos filtrar las señales de entrada del sistema para que se ajusten a estas

características. Una vez se disponga de las señales procesadas, el siguiente paso será la extracción de sus características, para lo que se ha generado una función que permite realizar el cálculo de los diferentes tipos de características seleccionadas.

Esta extracción de características, se hace utilizando ventanas de 3 segundos para la extracción, ya que el proceso, para este caso en concreto, no requiere una actualización inmediata del estado del usuario. Para evitar la posible pérdida de datos entre las ventanas de las señales, se ha realizado un overlapping de 1 segundo entre ventanas. Esto se representa de forma visual en la Figura 22, donde K tendría el valor de 3 segundos y $K-Q$ de 1 segundo.

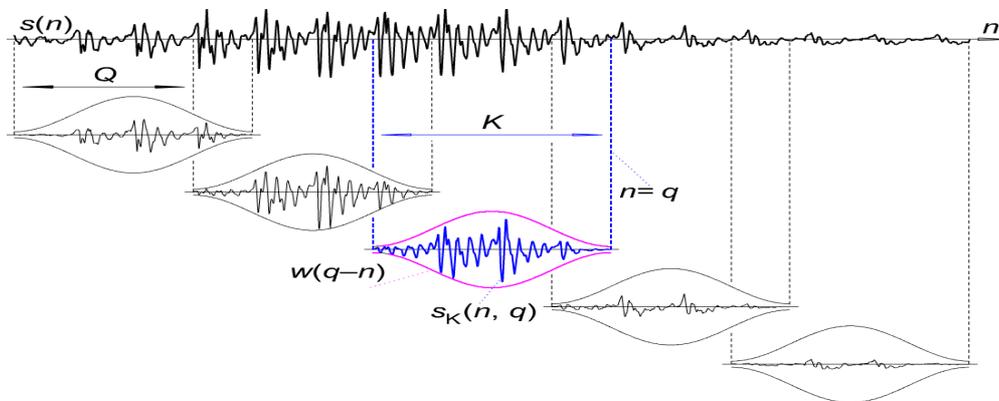


Figura 18. Overlapping de señales

Para poder clasificar estos datos extraídos de las señales EEG, ha sido necesario generar una base de datos a partir de las señales proporcionadas por DEAP dataset. Para esto, se han escogido los ocho electrodos más relevantes en el análisis de emociones como se ha comentado en el capítulo del Estado del Arte, es decir, los electrodos: Fp1, Fp2, F3, F4, P3, P4, O1 y O2. Estos equivalen a los canales número 1, 17, 3, 20, 11, 29, 14 y 32 de la base de datos.

Channel no.	Channel content
1	Fp1
2	AF3
3	F3
4	F7
5	FC5
6	FC1
7	C3
8	T7
9	CP5
10	CP1
11	P3
12	P7
13	PO3
14	O1
15	Oz
16	Pz
17	Fp2
18	AF4
19	Fz
20	F4
21	F8
22	FC6
23	FC2
24	Cz
25	C4
26	T8
27	CP6
28	CP2
29	P4
30	P8
31	PO4
32	O2

Figura 19. Canales de las EEG DEAP dataset

Una vez seleccionados los canales de las señales EEG de los electrodos que se desea analizar, se extraen sus características y almacenan en matrices de datos preparadas para entrenar los clasificadores que darán respuesta al estado de ánimo del conductor.

Estas matrices de datos, no siempre resultan ser iguales para los diferentes tipos de salida del sistema, lo que generaría un desequilibrio en el entrenamiento del clasificador. El uso de conjuntos desbalanceados en machine learning supone un problema ya que la mayoría de algoritmos son malos para manejarlos. Esto supone también problemas en la medición la precisión, ya que, si tenemos un número mucho mayor de datos de un conjunto que de otro, el algoritmo es posible que interprete la mayoría de datos como la clase mayoritaria.

Para solucionar este problema existen varias soluciones, entre ellas están el oversampling o el undersampling. Estas técnicas consisten en la eliminación de datos limitando todas

las clases a la minoritaria o el aumento de datos en los conjuntos al numero de datos de la clase mayoritaria, los cuales se generarían tomando como referencia los datos que se disponen.

En este caso, se ha utilizado la técnica de undersampling en las matrices de datos, que consiste como se ha comentado, en ajustar el tamaño de las matrices al más pequeño de estas, de forma que, aunque se pierdan datos, la clasificación pueda ser equilibrada y, por lo tanto, más efectiva.

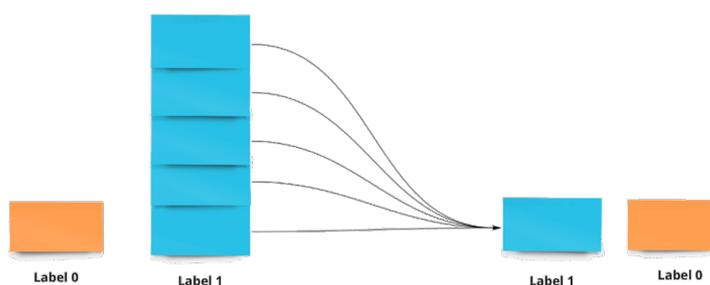


Figura 20. Técnica de undersampling

El siguiente paso entonces, es el entrenamiento del clasificador. Para este paso, como se ha comentado, se va a analizar el comportamiento de diferentes clasificadores con la herramienta de MATLAB Classification Learner, donde se prueban diferentes clasificadores evaluando la efectividad de cada uno de ellos.

Así pues, el esquema final del sistema se presenta de la forma que vemos en la Figura 25, las señales entran en el sistema y se filtran con un filtro de paso bajo entre 3-47 Hz ya que es el rango de frecuencia en el que se encuentran las señales EEG.



Figura 21. Diseño final del sistema

Cuando se obtenga una salida del sistema, que identifique si el conductor tiene altos o bajos los valores de valencia y excitación, se especificará en cuál de los cuatro estados de ánimo (Figura 11) se encuentra el conductor y se procederá a llevar a cabo el protocolo de actuación correspondiente.

4.4. Desarrollo del código

1.4.1. Cálculo de características

La primera función que se ha desarrollado en el apartado es la encargada de la extracción de características de las señales de los ocho electrodos seleccionados. Esta función tiene como entrada las ocho señales de los electrodos y el tipo de dato que se desea extraer y da como salida la matriz de datos deseada. (Figura 27)

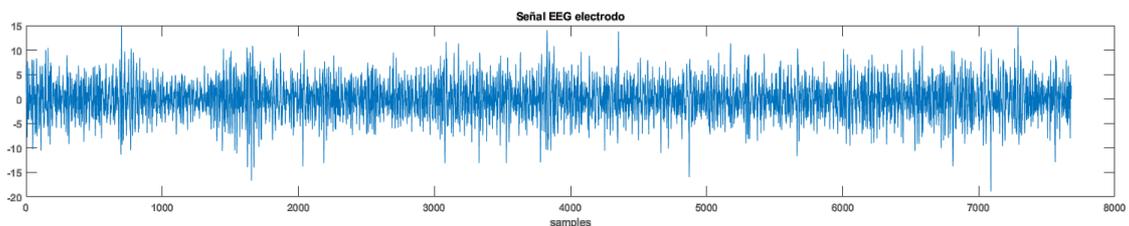


Figura 22. Señal EEG

Estas ocho señales se introducen en una misma matriz (*eeg_sf*) que es filtrada con un filtro de paso bajo de 3-47 Hz con una frecuencia de muestreo de 128 Hz. (Figura 28)

Así pues, cuando tenemos las 8 señales filtradas (matriz *eeg*), sacamos sus funciones intrínsecas con la función *emd* de MATLAB, ya que esta función es capaz de calcular la transformada de Descomposición Empírica en Modos y devolver las funciones intrínsecas de las señales. (Figura 29)

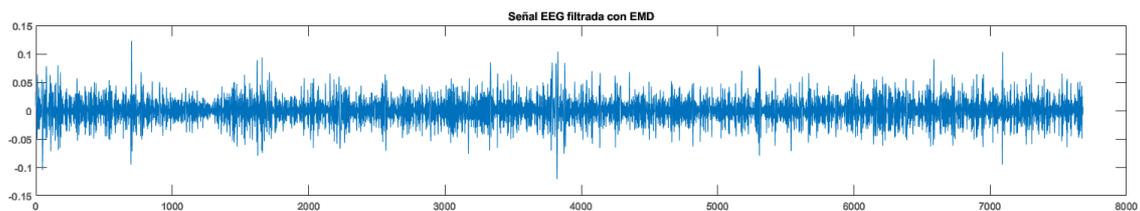


Figura 23. Señal filtrada con EMD

Las señales que se reciben en la función son de 60 segundos, por lo que, si se quiere extraer la información de la señal en ventanas de 3 segundos con un overlapping de 1 segundo, se requiere un bucle con 29 iteraciones. (Figura 31)

En cada una de estas iteraciones, primero se cargan las señales en los instantes en los que se desea, en este caso los relativos a 3 segundos, es decir $128\text{Hz} * 3 \text{ segundos} = 384$ muestras.

Cuando se hayan cargado las señales en los instantes de tiempo requeridos, se realizan los cálculos de las características que proporcionan 8 vectores de datos, uno por cada uno de las señales de los electrodos, que se almacenaran en la matriz de salida de datos.

Para realizar el overlapping, se ajusta en cada iteración del bucle de 29 iteraciones el valor *start*, que indica el comienzo de la carga de las señales en los instantes de tiempo. (Figura 32)

Para el cálculo de la primera de las características, es decir la primera diferencia de fase de las funciones se realiza atendiendo a la fórmula presentada en 3.3.2.2. (Figura 32)

El cálculo de la segunda característica, es decir de la primera diferencia de tiempo de las funciones intrínsecas, nuevamente, se calcula siguiendo la fórmula anteriormente explicada (3.3.2.3). (Figura 33)

Por último, la última de las características es la energía normalizada, esta se calcula a partir de la señal original además de la función intrínseca, por lo que también debemos cargar el conjunto de las señales sin ser filtradas con la función *emd* (*eeg_aux*). (Figura 34)

Para el cálculo de esta, se ha desarrollado el siguiente código, atendiendo a la fórmula expuesta anteriormente (3.3.2.4). (Figura 35)

1.4.2. Extracción de las características de las señales de DEAP dataset

Para extraer los datos específicos de señales de la base de datos DEAP dataset se ha generado una función en la que introduces el tipo de dato que se desea extraer, el participante y el video del que se desea obtener la señal y devuelve los datos calculados de las 8 señales de los electrodos seleccionados. (Figura 36)

Ya que los archivos en los que se encuentran las señales están en formato `s_participante_.mat`, la extracción de estos se ha realizado de la manera que se aprecia en la Figura 37.

Las señales de la base de datos, tienen una duración de 63 segundos (63 segundos * 128 Hz = 8064 samples), donde los 3 primeros segundos son vacíos, es decir, no interesantes para el análisis. Por esto, se cargan las señales a partir del segundo 3. (Figura 38)

De esta forma, se cargan las señales de los 8 electrodos desde el segundo 3 hasta el 63 de la matriz data que se incluye en los archivos de la base de datos, es decir, se cargan las señales de 60 segundos.

El último paso de esta función es extraer las características de estas señales con la función presentada al inicio del apartado. (Figura 39)

1.4.3. Generación de las matrices de datos a partir de las señales de DEAP dataset

En este apartado se realiza la extracción de las características relativas a cada estado emocional de las señales EEG de la base de datos de DEAP dataset, donde han sido clasificadas en función de su valencia y su excitación. Para esto se han generado dos vectores de participante y video para cada grupo de estado emocional. Estos vectores contienen las combinaciones de los participantes y videos donde la emoción ha sido clasificada en ese grupo. Por ejemplo, si en el vector `idParticipant1(33)` el valor es un 4 y en el vector `idVideo1(33)` el valor es un 10, significa que el participante 4 visualizando el video número 10, ha tenido un estado emocional del grupo 1. Esto se ha realizado para cada una de las tres características con los 4 grupos referidos a los 4 estados emocionales a identificar, donde el 1 es feliz, el 2 preocupado, el 3 triste y el 4 relajado.

Esta clasificación se ha realizado en función del valor de valencia y excitación proporcionado por DEAP dataset, donde un valor alto se considera a los valores por encima de 5 y un valor bajo es el que es igual o menor a 5. Así pues, el estado feliz tiene valores de valencia y excitación altos, preocupado valores de valencia bajos y excitación

alta, triste tiene ambos valores bajos, y, por último, relajado es indicado por una alta valencia y una baja excitación.

Para esta extracción se ha hecho uso de la función del apartado 4.4.2. introducida en un bucle que realiza tantas iteraciones como longitud de los arrays de `idParticipant` e `idVideo` sea. Cada uno de los datos extraídos en estas iteraciones, se almacenan en cuatro matrices, una para cada estado emocional, de la manera detallada en la Figura 40.

De la misma manera, se han calculado las características de los tipos 2 y 3 únicamente cambiando el tipo de dato a extraer en la función `ocho_electrodos_características`.

Así pues, el cálculo resultará en 12 matrices de datos, 4 por cada tipo de característica.

1.4.4. Entrenamiento del clasificador

Como es de suponer, no todas las matrices tienen las mismas dimensiones ya que hay algunas emociones que tienen más frecuencia que otras. Por esto, como se ha comentado antes, se va a usar la técnica de undersampling para agruparlas. (Figura 41)

Así pues, se realiza esta técnica en las 12 matrices de las que se dispone y se agrupan en función de su nivel de valencia, por una parte, y en función de su excitación por otra. De esta forma, se dispondrá de 8 características de cada tipo, es decir, de 24 características en total. Por ejemplo, en el caso de la clasificación de valencia la matriz se realizará de la manera mostrada en la Figura 42.

Dentro del machine learning, se va a utilizar el aprendizaje supervisado. Este es un método de análisis que consiste en el aprendizaje iterativo, lo que quiere decir que requiere unos datos de entrenamiento para ello.

Para la introducción de estos datos en la app Classification Learner, se debe asignar la salida de cada una de los datos de la matriz. Ya que el resultado final se divide en cuatro clases clasificables, el primer ensayo fue realizar un único clasificador que tuviera 4 salidas posibles. En este clasificador se obtuvo una precisión máxima de un 62 %, por lo que se optó por hacer la clasificación a partir de dos clasificadores binarios, uno para la evaluación de la valencia (alta o baja) y otro para la excitación (alta o baja).

Ya que esta herramienta permite ver cuál es el clasificador que mejor funciona frente a las matrices de datos de las que se dispone, se va a realizar el análisis de los dos clasificadores. (Figura 43)

Una vez se tenga la matriz que engloba los datos y sus respectivas salidas, se introduce en la app. En esta, se crea una nueva sesión donde introducimos la matriz que contiene los datos con sus respectivas salidas. (Figura 44)

En la aplicación, se puede seleccionar la opción de entrenar con todos los tipos de clasificadores disponibles, por lo que es útil para poder seleccionar el mejor de ellos. (Figura 45)

Una vez haya acabado este proceso de entrenamiento, aparecerán los valores de acuraccy (precisión) para cada uno de los clasificadores empleados. Cuando se haya seleccionado el mejor clasificador para cada una de las dos clasificaciones, se puede guardar en un archivo .mat y utilizar de la manera mostrada en la Figura 46, donde T y T2 son vectores de prueba cualesquiera.

En este punto, se obtienen los valores de valencia y excitación, por lo que el último paso es asignar el estado emocional de los cuatro existentes en función de ellos. Así pues, la señal de salida estará contenida en la señal *tipo_emocion* que tendrá el valor de 1 (feliz), 2(preocupado), 3(triste) o 4(relajado). (Figura 47)

4.5. Pruebas experimentales y análisis de resultados

Una vez extraídas las matrices de datos, se han realizado una serie de análisis para observarlas y sacar conclusiones sobre su comportamiento frente a la clasificación.

Se va a realizar un análisis del conjunto de datos que contiene las tres características. Para esto, se va a realizar una representación de su distribución, un análisis de su matriz de correlaciones y por último un análisis de componentes principales (PCA). Con esto análisis, se puede extraer información de la relevancia de cada una de las características y lo relacionadas que están entre ellas. El análisis PCA permite visualizar el porcentaje de cada una de las características para su clasificación, lo que resulta útil en conjuntos de numerosas características ya que podrían eliminarse las características menos relevantes.

La representación se realiza por medio de la función *boxplot* de MATLAB, y representa cajas por cada columna de la matriz de datos. La marca dentro de la caja indica la mediana y los trazos se extienden hasta los datos más extremos sin ser considerados outliers (datos atípicos), estos últimos se representan con cruces.

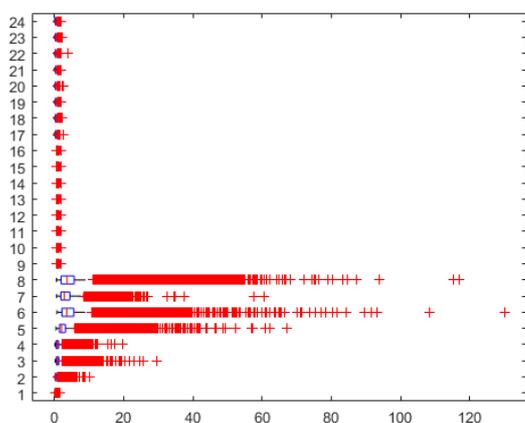


Figura 24. Representación matriz de datos

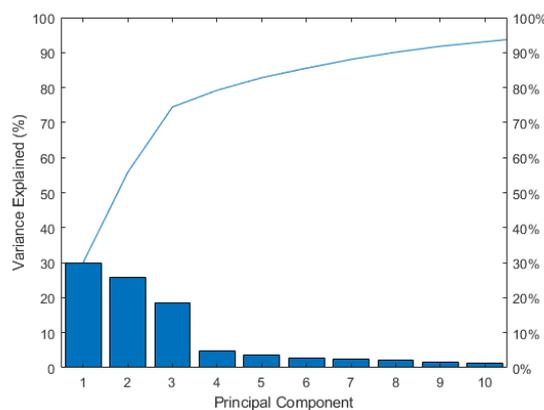


Figura 25. Análisis PCA matriz de datos

CT = 24x24 table

	C1	C2	C3	C4	C5	C6
1	1.0000	0.8786	0.8423	0.5690	0.8557	0.8055
2	0.8786	1.0000	0.9781	0.6260	0.9459	0.9211
3	0.8423	0.9781	1.0000	0.6882	0.9383	0.9659
4	0.5690	0.6260	0.6882	1.0000	0.5641	0.7286
5	0.8557	0.9459	0.9383	0.5641	1.0000	0.9259
6	0.8055	0.9211	0.9659	0.7286	0.9259	1.0000
7	0.5381	0.5225	0.5669	0.8314	0.5446	0.6298
8	0.7557	0.7576	0.7624	0.4695	0.8900	0.7941
9	0.2115	-0.0737	-0.0775	0.0362	-0.0286	-0.0353

Figura 26. Matriz correlación características

De este análisis podemos sacar algunas conclusiones. En primer lugar, se puede apreciar en la distribución que la distribución de los datos se encuentra en rangos bastante claros, lo que resulta beneficioso para la clasificación.

Con respecto al análisis de componentes principales (PCA), se puede ver como a partir de la décima característica se alcanza casi el 95% de los datos, por lo que se podría reducir el número de características seleccionándose las 10 primeras para el nuevo análisis.

Por último, la matriz de correlaciones indica la correlación de todas las características entre sí, por lo que cuanto más bajos sean estos valores mejor, ya que indicará que son más relevantes.

Tras este análisis de la matriz de datos, se han realizado dos pruebas con los clasificadores, una de ellas sin aplicar los resultados obtenidos con análisis PCA y la segunda con este. Para esto, se han introducido los valores en la aplicación Classification Learner como se ha explicado anteriormente. La posibilidad de realizar este análisis de clasificadores, permite aumentar la precisión de clasificación seleccionando el mejor clasificador para cada uno de ambos casos.

En primer lugar, se ha introducido la matriz para la clasificación de la valencia, donde la salida del clasificador indicará si se ha obtenido un valor de valencia alto o bajo. Para este análisis se han obtenido los siguientes resultados para cada uno de los clasificadores:

1.1 ☆ Tree Last change: Fine Tree	Accuracy: 59.3% 324 features (PCA on)	1.1 ☆ Tree Last change: Fine Tree	Accuracy: 61.8% 2424 features
1.2 ☆ Tree Last change: Medium Tree	Accuracy: 57.5% 324 features (PCA on)	1.2 ☆ Tree Last change: Medium Tree	Accuracy: 58.5% 2424 features
1.3 ☆ Tree Last change: Coarse Tree	Accuracy: 56.0% 324 features (PCA on)	1.3 ☆ Tree Last change: Coarse Tree	Accuracy: 57.5% 2424 features
1.4 ☆ Linear Discriminant Last change: Linear Discriminant	Accuracy: 53.7% 324 features (PCA on)	1.4 ☆ Linear Discriminant Last change: Linear Discriminant	Accuracy: 57.3% 2424 features
1.5 ☆ Quadratic Discriminant Last change: Quadratic Discriminant	Accuracy: 52.0% 324 features (PCA on)	1.5 ☆ Quadratic Discriminant Last change: Quadratic Discriminant	Accuracy: 56.4% 2424 features
1.6 ☆ Logistic Regression Last change: Logistic Regression	Accuracy: 53.7% 324 features (PCA on)	1.6 ☆ Logistic Regression Last change: Logistic Regression	Accuracy: 57.4% 2424 features
1.7 ☆ Naive Bayes Last change: Gaussian Naive Bayes	Accuracy: 52.0% 324 features (PCA on)	1.7 ☆ SVM Last change: Linear SVM	Accuracy: 57.8% 2424 features
1.8 ☆ Naive Bayes Last change: Kernel Naive Bayes	Accuracy: 57.3% 324 features (PCA on)	1.8 ☆ SVM Last change: Quadratic SVM	Accuracy: 63.2% 2424 features
1.9 ☆ SVM Last change: Linear SVM	Accuracy: 50.2% 324 features (PCA on)	1.9 ☆ SVM Last change: Cubic SVM	Accuracy: 63.2% 2424 features
1.10 ☆ SVM Last change: Quadratic SVM	Accuracy: 49.9% 324 features (PCA on)	1.10 ☆ SVM Last change: Fine Gaussian SVM	Accuracy: 65.9% 2424 features
1.11 ☆ SVM Last change: Cubic SVM	Accuracy: 49.5% 324 features (PCA on)	1.11 ☆ SVM Last change: Medium Gaussian SVM	Accuracy: 64.6% 2424 features
1.12 ☆ SVM Last change: Fine Gaussian SVM	Accuracy: 59.3% 324 features (PCA on)	1.12 ☆ SVM Last change: Coarse Gaussian SVM	Accuracy: 58.8% 2424 features
1.13 ☆ SVM Last change: Medium Gaussian SVM	Accuracy: 55.5% 324 features (PCA on)	1.13 ☆ KNN Last change: Fine KNN	Accuracy: 62.5% 2424 features
1.14 ☆ SVM Last change: Coarse Gaussian SVM	Accuracy: 53.9% 324 features (PCA on)	1.14 ☆ KNN Last change: Medium KNN	Accuracy: 63.2% 2424 features
1.15 ☆ KNN Last change: Fine KNN	Accuracy: 68.0% 324 features (PCA on)	1.15 ☆ KNN Last change: Coarse KNN	Accuracy: 62.5% 2424 features
1.16 ☆ KNN Last change: Medium KNN	Accuracy: 61.6% 324 features (PCA on)	1.16 ☆ KNN Last change: Cosine KNN	Accuracy: 63.4% 2424 features
1.17 ☆ KNN Last change: Coarse KNN	Accuracy: 62.3% 324 features (PCA on)	1.17 ☆ KNN Last change: Cubic KNN	Accuracy: 62.5% 2424 features
1.18 ☆ KNN Last change: Cosine KNN	Accuracy: 58.3% 324 features (PCA on)	1.18 ☆ KNN Last change: Weighted KNN	Accuracy: 64.6% 2424 features
1.19 ☆ KNN Last change: Cubic KNN	Accuracy: 61.7% 324 features (PCA on)	1.19 ☆ Ensemble Last change: Boosted Trees	Accuracy: 61.1% 2424 features
1.20 ☆ KNN Last change: Weighted KNN	Accuracy: 60.9% 324 features (PCA on)	1.20 ☆ Ensemble Last change: Bagged Trees	Accuracy: 68.9% 2424 features
1.21 ☆ Ensemble Last change: Boosted Trees	Accuracy: 59.1% 324 features (PCA on)	1.21 ☆ Ensemble Last change: Subspace Discriminant	Accuracy: 56.3% 2424 features
1.22 ☆ Ensemble Last change: Bagged Trees	Accuracy: 61.1% 324 features (PCA on)	1.22 ☆ Ensemble Last change: Subspace KNN	Accuracy: 68.3% 2424 features
1.23 ☆ Ensemble Last change: Subspace Discriminant	Accuracy: 53.6% 324 features (PCA on)	1.23 ☆ Ensemble Last change: RUSBoosted Trees	Accuracy: 58.5% 2424 features
1.24 ☆ Ensemble Last change: Subspace KNN	Accuracy: 55.5% 324 features (PCA on)		
1.25 ☆ Ensemble Last change: RUSBoosted Trees	Accuracy: 57.5% 324 features (PCA on)		

Figura 27. Análisis de clasificadores para la valencia alta o baja (con PCA izquierda, sin PCA derecha)

Como se puede observar en la Figura 51, el mejor clasificador ha sido el obtenido utilizando *Bagged Trees* sin análisis PCA, con una precisión de 68.9%, por lo que es el clasificador que se va a utilizar en el proyecto para la clasificación de la valencia.

El clasificador Bagged Trees consiste en un algoritmo de machine learning basado en el uso de varios árboles de decisión, por lo que permite aprovechar el rendimiento de los diferentes modelos. Los arboles de decisión siguen estructuras ramificadas en las que se fabrican construcciones lógicas para categorizar una serie de condiciones sucesivas.

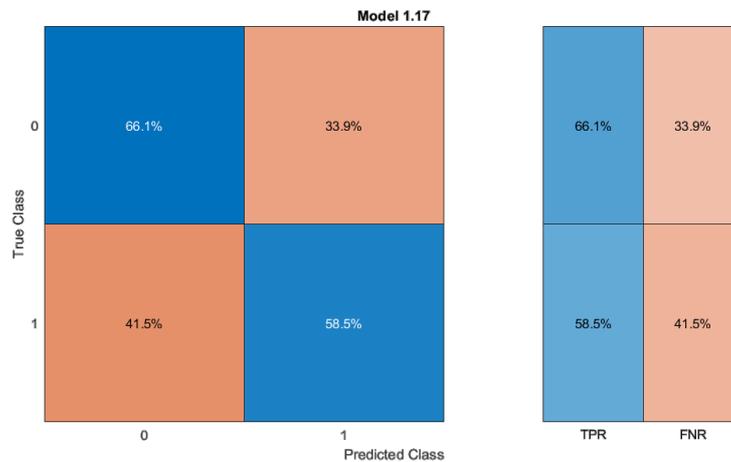


Figura 28. Matriz de confusión de clasificación valencia con PCA

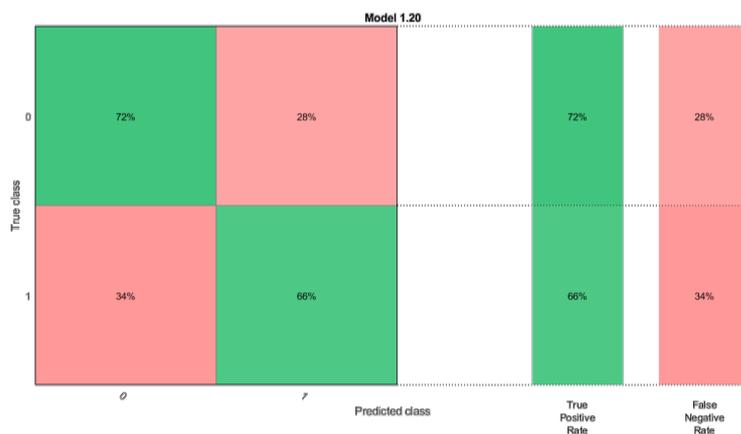


Figura 29. Matriz de confusión de clasificación valencia sin PCA

En las matrices de confusión se puede apreciar el porcentaje de error y acierto del clasificador en la fase de validación. Con esto llegamos a la misma conclusión, la clasificación sin uso del análisis PCA es más efectiva puesto que tiene valores de precisión más elevados.

Nuevamente, para la clasificación de la excitación se ha realizado el mismo proceso, donde se proporciona como salida el valor de la excitación, es decir, si esta es alta o baja.



1.1 ☆ Tree Last change: Fine Tree	Accuracy: 62.6% 3/24 features (PCA on)	1.1 ☆ Tree Last change: Fine Tree	Accuracy: 63.9% 24/24 features
1.2 ☆ Tree Last change: Medium Tree	Accuracy: 60.2% 3/24 features (PCA on)	1.2 ☆ Tree Last change: Medium Tree	Accuracy: 61.9% 24/24 features
1.3 ☆ Tree Last change: Coarse Tree	Accuracy: 57.9% 3/24 features (PCA on)	1.3 ☆ Tree Last change: Coarse Tree	Accuracy: 56.9% 24/24 features
1.4 ☆ Linear Discriminant Last change: Linear Discriminant	Accuracy: 56.9% 3/24 features (PCA on)	1.4 ☆ Linear Discriminant Last change: Linear Discriminant	Accuracy: 57.8% 24/24 features
1.5 ☆ Quadratic Discriminant Last change: Quadratic Discriminant	Accuracy: 53.4% 3/24 features (PCA on)	1.5 ☆ Quadratic Discriminant Last change: Quadratic Discriminant	Accuracy: 61.1% 24/24 features
1.6 ☆ Logistic Regression Last change: Logistic Regression	Accuracy: 57.0% 3/24 features (PCA on)	1.6 ☆ Logistic Regression Last change: Logistic Regression	Accuracy: 57.8% 24/24 features
1.7 ☆ Naive Bayes Last change: Gaussian Naive Bayes	Accuracy: 53.0% 3/24 features (PCA on)	1.7 ☆ SVM Last change: Linear SVM	Accuracy: 58.3% 24/24 features
1.8 ☆ Naive Bayes Last change: Kernel Naive Bayes	Accuracy: 56.9% 3/24 features (PCA on)	1.8 ☆ SVM Last change: Quadratic SVM	Accuracy: 63.6% 24/24 features
1.9 ☆ SVM Last change: Linear SVM	Accuracy: 50.2% 3/24 features (PCA on)	1.9 ☆ SVM Last change: Cubic SVM	Accuracy: 62.7% 24/24 features
1.10 ☆ SVM Last change: Quadratic SVM	Accuracy: 49.9% 3/24 features (PCA on)	1.10 ☆ SVM Last change: Fine Gaussian SVM	Accuracy: 68.2% 24/24 features
1.11 ☆ SVM Last change: Cubic SVM	Accuracy: 50.3% 3/24 features (PCA on)	1.11 ☆ SVM Last change: Medium Gaussian SVM	Accuracy: 66.3% 24/24 features
1.12 ☆ SVM Last change: Fine Gaussian SVM	Accuracy: 61.2% 3/24 features (PCA on)	1.12 ☆ SVM Last change: Coarse Gaussian SVM	Accuracy: 59.4% 24/24 features
1.13 ☆ SVM Last change: Medium Gaussian SVM	Accuracy: 58.1% 3/24 features (PCA on)	1.13 ☆ KNN Last change: Fine KNN	Accuracy: 64.1% 24/24 features
1.14 ☆ SVM Last change: Coarse Gaussian SVM	Accuracy: 52.9% 3/24 features (PCA on)	1.14 ☆ KNN Last change: Medium KNN	Accuracy: 65.1% 24/24 features
1.15 ☆ KNN Last change: Fine KNN	Accuracy: 60.1% 3/24 features (PCA on)	1.15 ☆ KNN Last change: Coarse KNN	Accuracy: 63.6% 24/24 features
1.16 ☆ KNN Last change: Medium KNN	Accuracy: 64.3% 3/24 features (PCA on)	1.16 ☆ KNN Last change: Cosine KNN	Accuracy: 64.5% 24/24 features
1.17 ☆ KNN Last change: Coarse KNN	Accuracy: 65.4% 3/24 features (PCA on)	1.17 ☆ KNN Last change: Cubic KNN	Accuracy: 63.9% 24/24 features
1.18 ☆ KNN Last change: Cosine KNN	Accuracy: 61.0% 3/24 features (PCA on)	1.18 ☆ KNN Last change: Weighted KNN	Accuracy: 66.2% 24/24 features
1.19 ☆ KNN Last change: Cubic KNN	Accuracy: 64.2% 3/24 features (PCA on)	1.19 ☆ Ensemble Last change: Boosted Trees	Accuracy: 65.5% 24/24 features
1.20 ☆ KNN Last change: Weighted KNN	Accuracy: 63.5% 3/24 features (PCA on)	1.20 ☆ Ensemble Last change: Bagged Trees	Accuracy: 72.0% 24/24 features
1.21 ☆ Ensemble Last change: Boosted Trees	Accuracy: 63.1% 3/24 features (PCA on)	1.21 ☆ Ensemble Last change: Subspace Discriminant	Accuracy: 55.7% 24/24 features
1.22 ☆ Ensemble Last change: Bagged Trees	Accuracy: 63.4% 3/24 features (PCA on)	1.22 ☆ Ensemble Last change: Subspace KNN	Accuracy: 71.4% 24/24 features
1.23 ☆ Ensemble Last change: Subspace Discriminant	Accuracy: 57.0% 3/24 features (PCA on)	1.23 ☆ Ensemble Last change: RUSBoosted Trees	Accuracy: 62.0% 24/24 features
1.24 ☆ Ensemble Last change: Subspace KNN	Accuracy: 56.6% 3/24 features (PCA on)		
1.25 ☆ Ensemble Last change: RUSBoosted Trees	Accuracy: 60.2% 3/24 features (PCA on)		

Figura 30. Análisis de clasificadores para la excitación alta o baja (con PCA izquierda, sin PCA derecha)

Los resultados para esta clasificación son los presentados en la Figura 53. Como se puede apreciar, el valor de accuracy más alto es el obtenido nuevamente en el clasificador *Bagged Trees* sin análisis PCA, con un 72 %, por lo que es el que se va a utilizar en el proyecto.

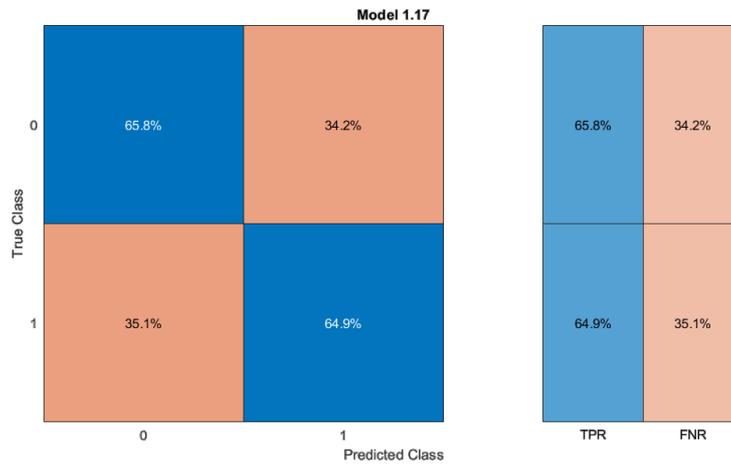


Figura 31. Matriz de confusión de clasificación excitación con PCA

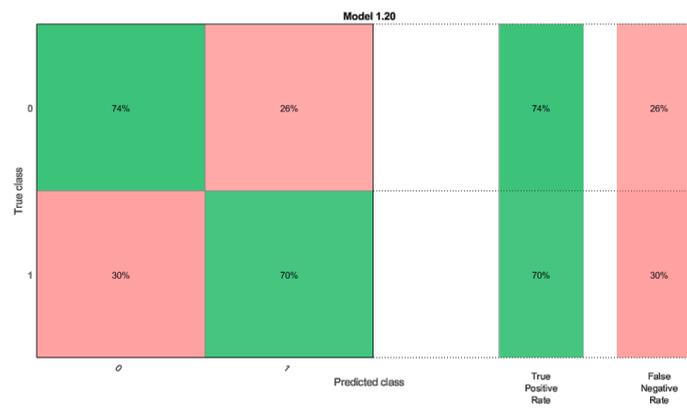


Figura 32. Matriz de confusión de clasificación excitación sin PCA

Nuevamente, llegamos a la misma conclusión analizando las matrices de confusión, la clasificación sin uso del análisis PCA es más efectiva puesto que tiene valores de precisión más elevados.

	Accuracy valencia	Accuracy excitación
Resultados obtenidos en el artículo de referencia [1]	69%	72%
Resultados obtenidos con el análisis PCA	62.3%	65.4%
Resultados obtenidos con el análisis PCA	68.9%	72%

Tabla 1. Comparativa de resultados de precisión del clasificador

Estos valores escogidos (sin PCA) son casi iguales a los obtenidos en el artículo [1], donde se obtiene un 69 % para la valencia y un 72 % para la excitación por medio de

clasificadores SVM. Por esto, se considera que son buenos resultados y, por lo tanto, son los clasificadores que se van a utilizar en el sistema.

Por último, con respecto al análisis del coste computacional, se ha analizado por una parte el tiempo de ejecución total, el tiempo de cálculo de características y el tiempo de predicción con los clasificadores. El tiempo de ejecución de la extracción de las características es de 0.3 segundos y el tiempo que tarda el predecir el tipo de emoción el clasificador es de 0.2 segundos, lo que resulta en un tiempo de ejecución del sistema total de 0.5 segundos. Esto resulta adecuado ya que en este tipo de aplicación en el que no se requiere una respuesta inmediata, estos valores son suficientes para la detección de emociones en un tiempo razonable.

4.6. Desarrollo de protocolos de actuación

Como se ha detallado anteriormente en la memoria, cada estado de ánimo tiene asignado un modo de conducción, este modo de conducción se ajusta con los parámetros de suspensión de modelos vehiculares creados anteriormente en AIC para el simulador.

Para observar el comportamiento de cada uno de los modelos de suspensión se han generado sus curvas de tracción y compresión de la parte frontal y trasera del vehículo ya que la masa del vehículo se reparte en estas zonas.

El primer modo de conducción es el modelo por defecto de suspensión del vehículo, es decir, el equivalente al estado emocional de calma (Figura 57).

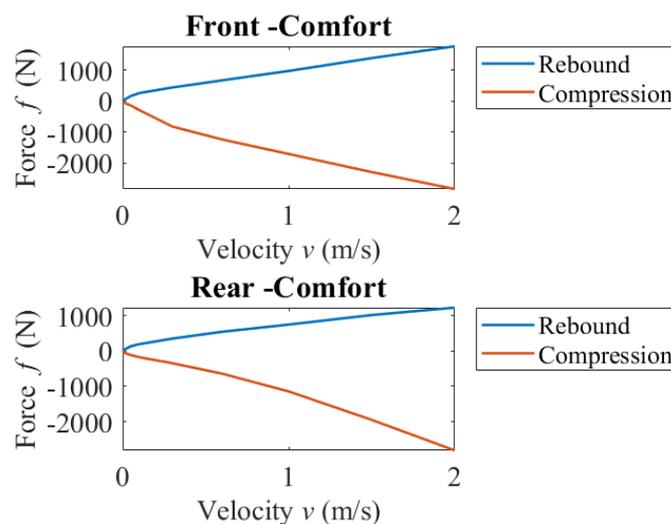


Figura 33. Curvas de suspensión modo confort

Aquí podemos ver que los valores normales entre los que está la fuerza de tracción del coche es de 1500 N y los de compresión de -2500 N.

El modelo de suspensión utilizado en el modo sport es el equivalente al estado emocional de felicidad. Las curvas de este modelo de suspensión se representan en la Figura 58.

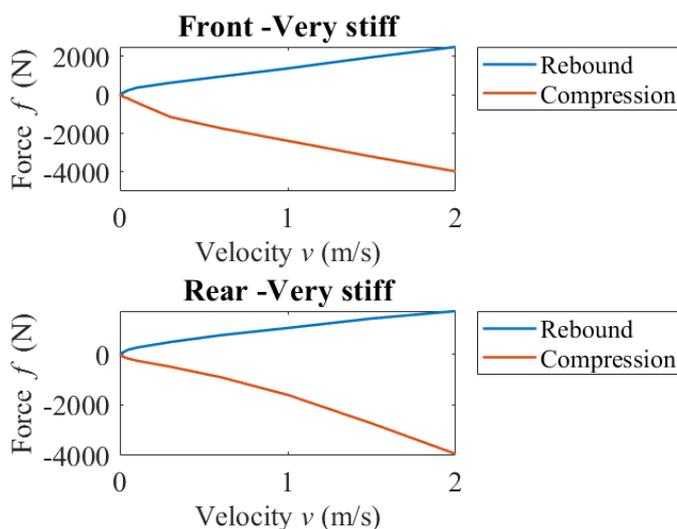


Figura 34. Curvas de suspensión modo sport

Como se puede apreciar en estas curvas de suspensión, el valor de la fuerza de tracción y compresión son del doble, por lo que es una suspensión más dura.

Para el modelo de suspensión del modo de conducción equivalente al estado emocional de tristeza, se propone un modelo de suspensión más blanda. Esto quiere decir que los valores de tracción y suspensión de esta serán más pequeños que en la suspensión por defecto del vehículo (Figura 59).

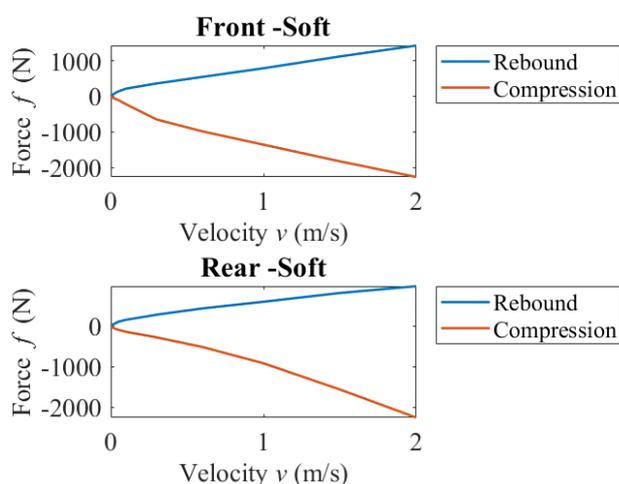


Figura 35. Curvas de suspensión modo personalizado tristeza

Por último, para el modo de conducción asignado al estado emocional de preocupación, se ha seleccionado un modelo de suspensión más duro que el modelo por defecto, pero

sin llegar a la magnitud de fuerzas del modo sport, es decir, un modelo de suspensión intermedio entre el modo sport y modo confort.

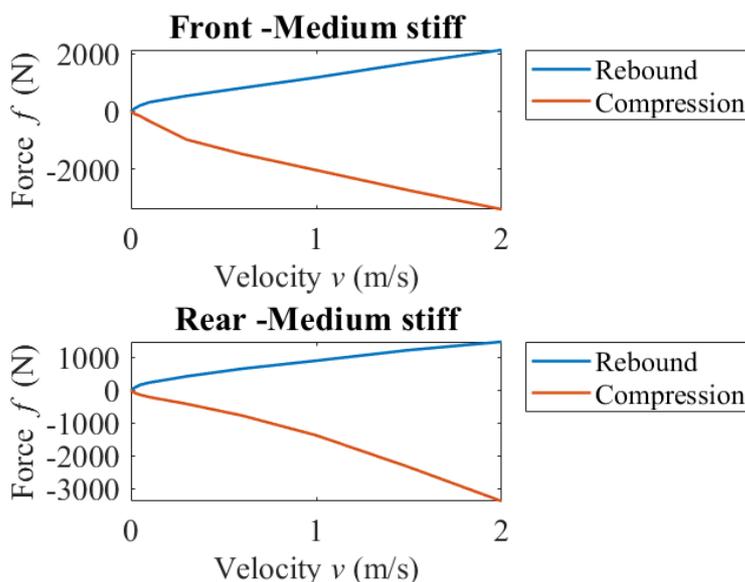


Figura 36. Curvas de suspensión modo personalizado preocupación

Como se puede observar en la Figura 60, los valores de tracción llegan a 2000 N y la fuerza de compresión a -3000 N, lo que equivale a valores medios entre ambos modos como se ha comentado.

4.7. Validación y test

La finalidad de este apartado, es llevar a cabo el testeo del sistema en su conjunto para poder validar su funcionamiento. Para esto se ha generado un código en el que se introducirían las señales que entrarían en el sistema si fuera implementado de en un dispositivo real.

Como ejemplo, se ha seleccionado las señales del participante 5 en el video 12, que sabemos que pertenece al grupo de estado emocional 2, es decir al estado emocional de preocupación. Ya que estas señales pertenecen a la base de datos se utiliza la función *ocho_características_electrodos*, de no ser así, se emplearía la función de extracción de características como se ve comentado en la Figura 61, donde las señales data1, data2, data3, data4, data5, data6, data7 y data8 pertenecen a las señales de los 8 electrodos medidos por el dispositivo.

Cuando se ejecuta el script, se obtiene como resultado que el tipo de emoción es el tipo 2 (Figura 62).

El resultado obtenido es correcto por lo que, tras hacer más verificaciones, se puede concluir con que el sistema funciona correctamente.

4.8. Implementación hardware

Para implementar este sistema en el simulador, debemos controlarlo a través de un ordenador auxiliar. Este ordenador, que en este caso está dentro del equipo del simulador con el fin de implementar este tipo de funciones, se comunicará con el simulador por UDP como se detalló en el apartado 3.3.8.

Para esto, se crea un modelo de Simulink en el ordenador auxiliar con un bloque UDP para poder enviar la salida del sistema al esquema de Simulink que controla el simulador (Figura 21).

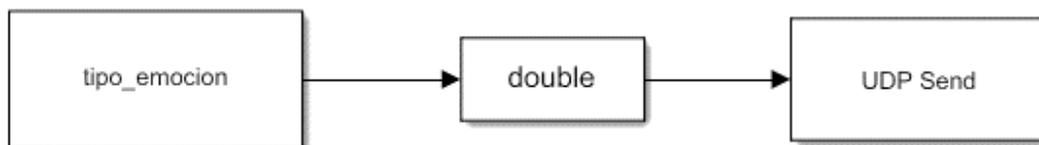


Figura 37. Esquema Simulink UDP send

La variable *tipo_emocion* es la variable de salida del sistema y tiene cuatro valores posibles en función del tipo de emoción que se detecte: 1, 2, 3 o 4.

El bloque de *UDP Send* se configura con la IP del ordenador que controla el simulador, en este caso es la dirección 10.1.90.15 y le asignamos un puerto que no esté siendo utilizado.

Tanto la variable que envía el sistema como el bloque UDP que recibe deben de estar configurados para que el tipo de variable que se transfiere sea el mismo, por esto, se ha añadido un bloque *convert*, a tipo en este caso *double*, en el esquema de envío.

El bloque de recepción UDP ha de ser incluido en el esquema de Simulink del simulador, dentro del bloque encargado de la dinámica vehicular del simulador (Car Maker), más concretamente en la sección dedicada a asignar los parámetros a la suspensión del coche, por lo que debemos llevar la señal de salida que se recibe hasta este bloque para poder trabajar con ella.

Para recibir los datos, se añade en la ubicación indicada, el bloque *UDP Receiver* y se configura para que reciba la información en el mismo puerto que se ha especificado en el bloque de envío UDP. Además, hay que especificar el tipo y tamaño máximo del mensaje, en este caso, al tratarse de variable tipo *double* es de 1 byte. Por último, se debe introducir el periodo de muestreo, que es 0.001 segundos ya que corresponde a la frecuencia a la que trabaja el simulador (1000 Hz).



Figura 38. Esquema Simulink UDP receive

En este caso como se ha detallado, el sistema puede proporcionar cuatro salidas diferentes que representan los cuatro estados emocionales analizados. Cada uno de ellos, tiene asignado el modelo de coche que se ha descrito en el apartado 4.6., por lo que una vez que el valor de salida se reciba en el esquema de Simulink del simulador, se configurará un modelo de coche u otro en función de la salida.

Para esto se ha añadido un bloque que contiene una función de MATLAB que da directamente como salida los parámetros de suspensión que configurarán la dinámica del simulador. Esta función carga los modelos de suspensión del vehículo en función del valor de entrada que recibe, es decir, en función del tipo de emoción que detecta el sistema. (Figura 65)

Una vez realizadas estas modificaciones, se hace una prueba para validar la correcta comunicación entre el sistema de monitorización cerebral y la plataforma de simulación. En esta, se ha configurado en el envío una señal que cambia de valor entre 1, 2, 3 y 4 (Figura 66).

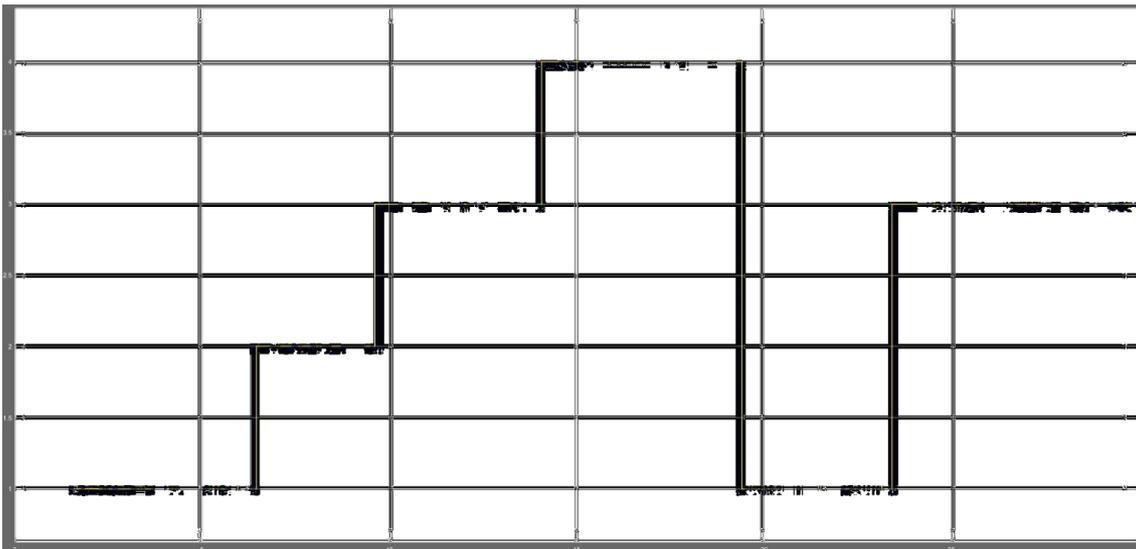


Figura 39. Test comunicación UDP

eman ta zabal zazu



Universidad del País Vasco Euskal Herriko Unibertsitatea

BILBOKO
INGENIARITZA
ESKOLA

Como se puede ver en la imagen, la comunicación UDP pierde algunos datos en el proceso como se explicó, lo que en este tipo de aplicación no es relevante ya que el cambio de modo de conducción no debe ser instantáneo.

Capítulo 5. EJEMPLO DE APLICACIÓN

En este capítulo, se va a plantear una situación en la que el uso de este tipo de sistema podría ser de utilidad, llegando incluso a evitar un accidente.

La Dirección General de Tráfico declara que año tras año el factor que más accidentes causa es el debido a las distracciones al volante. Muchas de estas distracciones son debidas a un motivo previo, es decir, un acontecimiento que ha alterado nuestro estado emocional y, por lo tanto, a nuestra conducción.

Una situación habitual es estar preocupado o estresado debido a algún acontecimiento ocurrido en el trabajo, en la universidad o incluso en la vida personal. Si se plantea la situación en la que una persona sale del trabajo con más estrés de lo habitual puesto que un proyecto no ha salido como debía, esto va a afectar directamente en su conducción de vuelta a casa.



Figura 40. Conductor en una situación de estrés

Esta persona se subirá en su vehículo con la mente pensando en el proyecto, lo que resulta en una distracción en la conducción. Por esta causa, en el camino de vuelta, sin fijarse demasiado en la carretera, acelera demasiado y realiza una conducción agresiva, lo que le podría causar un accidente, por ejemplo, por no haber visto una señal de stop en un cruce e ir demasiado deprisa como para frenar a tiempo.

Aunque el sistema planteado no evitaría directamente el accidente, un prototipo futuro podría hacerlo si los protocolos de actuación abarcasen el control total del vehículo. De esta forma, el sistema detectaría que el sujeto se encuentra en un estado de preocupación por lo que el vehículo se encontraría alerta frente a situaciones que pudieran poner en riesgo al conductor y el entorno que lo rodea y, en este caso, habría reducido la velocidad para que no rebasase la establecida y le habría dado tiempo a frenar.

En una situación similar a esta, en la que el conductor entra en un estado de distracción, el sistema planteado podría ayudar de modo que una conducción más adecuada al estado del conductor, apaciguase su preocupación mejorando el confort en el vehículo y calmase la agresividad al volante, lo que a priori no evitaría un accidente, pero podría prevenirlo.



Capítulo 6. CONCLUSIONES Y TRABAJO FUTURO

Para la finalización del proyecto realizado, es necesario realizar un análisis del cumplimiento de los objetivos planteados al inicio de la memoria y la extracción de conclusiones en función de los resultados conseguidos en este punto.

Tras llevar a cabo el desarrollo del proceso siguiendo el artículo [1], donde se clasifican las señales EEG en función de su valencia y excitación haciendo uso de la base de datos DEAP dataset, se podría concluir con que los resultados que se han obtenido son acordes a la línea de la investigación que se ha seguido. Aunque no se han podido utilizar las mismas herramientas que se han empleado en el artículo mencionado, los resultados resultan muy aproximados en cuanto a porcentajes, logrando aproximadamente un 70% de precisión de las clasificaciones.

En cuanto a los objetivos planteados inicialmente, se ha logrado desarrollar un sistema de monitorización del conductor usuario para analizar el estado emocional a través de ondas cerebrales. El sistema extrae las características de las señales en distintas zonas del cerebro y las clasifica en función de la valencia y la excitación para determinar en qué estado emocional se encuentra el usuario.

En primer lugar, se ha realizado un estudio de investigación en el tema a tratar y se han investigado los elementos de los que dispone el software, además de un estudio de los posibles dispositivos hardware de medición de señales cerebrales que podrían usarse en este proyecto.

Una vez se ha realizado el estudio se ha elaborado el diseño en el que se ha basado el proyecto. El siguiente paso entonces, ha sido generar el código en MATLAB que ha logrado el funcionamiento deseado del sistema.

Cuando se ha acabado de generar el código, se han realizado los testes y validaciones para comprobar que funcione de la manera esperada detectando así los cuatro tipos de emociones posibles, felicidad, preocupación, tristeza o relajación.

Por último, ya que finalmente no se ha adquirido el hardware, no se ha podido implementar el sistema en la realidad, pero se han establecido los protocolos de actuación en el simulador de conducción de AIC.

Finalmente, los resultados que se han obtenido dejan ver una vez más, que existe una relación entre las emociones del ser humano y las señales del cerebro, ya que con estas se es capaz de identificar los diferentes estados emocionales a partir de estas EEG.

La implementación de la inteligencia artificial en vehículos ha tenido una vertiginosa evolución en estos últimos años, siendo capaces estos de llegar a niveles de autonomía en los que ni siquiera es necesario un ser humano para controlarlos. Sin embargo, un gran paso a dar es el control de los vehículos en función del estado del conductor, siendo así



dependientes de este, pero aportando mayor control puesto que se pueden analizar variables que ni el conductor por sí mismo percibe.

Este tipo de sistemas de monitorización del conductor de un vehículo, en este caso cerebralmente, resultan de gran utilidad para mejorar la calidad de vida cotidiana de las personas. Por esto, ya que en este proyecto finalmente no ha sido posible la adquisición de un hardware, se propone como trabajo futuro la implementación de estos sistemas en un dispositivo de medición de señales cerebrales para llevar a cabo las validaciones y pruebas requeridas que ayuden a continuar con el desarrollo de este tipo de sistemas.

Se propone, además, la posibilidad de desarrollar más profundamente los diferentes protocolos de actuación en los que el sistema también sea capaz de evitar directamente accidentes, por ejemplo, haciendo uso del sistema de frenado o reduciendo la velocidad.

Cabe destacar también, la necesidad de continuar explorando la extracción de diferentes tipos de características que sean capaces de lograr porcentajes entorno a un 80% de precisión en la clasificación de las emociones para que puedan implementarse de manera más fiable en dispositivos.

Puesto que esta es un área en continuo desarrollo y evaluación, se espera que este proyecto sirva de base para posteriores proyectos que sigan las mismas líneas de trabajo y lo lleven más allá implementando todo tipo de mejoras, para que en un futuro llegue a ser un sistema implementable en todo tipo de vehículo.

BIBLIOGRAFÍA

- [1] Z. Ning, Z. Ying, Li Tong, Z. Hanming y Y. Bin. «Emotion recognition from EEG signals using multidimensional information in EMD domain,» Hindawi, vol. 2017, pp. 1-9, 2017.
- [2] APD España. «Aplicaciones reales de la inteligencia artificial en la medicina», 23 de febrero de 2021.
- [3] «CUESTIONES DE SEGURIDAD VIAL. Edición 2020.», s. f., 556. Fundación Carlos Slim. «Distracciones al conducir – Seguridad Vial», 31 de agosto de 2016.
- [4] Carmen Contreras, Woocar. «Nuestras emociones y nuestra forma de manejar», 27 de febrero de 2018.
- [5] AIC - Automotive Intelligence Center. « Virtual Development Center».
- [6] L. Li, K. Ota, and M. Dong, _Humanlike driving: Empirical decision-making system for autonomous vehicles, IEEE Transactions on Vehicular Technology, vol. 67, no. 8, pp. 6814_6823, aug 2018.
- [7] Q. Yuan, Y. Gao, and Y. Li, “Suppose future traffic accidents based on development of self-driving vehicles,” in Man-Machine-Environment System Engineering. New York, NY, USA: Springer, 2016, pp. 253–261.
- [8] Greco, Carolina. «LAS EMOCIONES POSITIVAS: SU IMPORTANCIA EN EL MARCO DE LA PROMOCIÓN DE LA SALUD MENTAL EN LA INFANCIA», 2010, 14.
- [9] «Sistema límbico: la parte emocional del cerebro», 23 de agosto de 2016.
- [10] Pal, Mayukha, Neeraj, y Prasanta K. Panigrahi. «Coupled Oscillator Dynamics in Brain EEG Signals: Manifestation of synchronization and Across Frequency Energy Exchange by Neutral Turbulence». *IEEE Access* 10 (2022): 20445-54.
- [11] E. R. Kandel, J. H. Schwartz, T. M. Jessell. «Principles of Neural Science», Mc Graw Hill.
- [12] Vivar, Sergio, Maria Abud, Gabriel Vásquez, Ulises Juárez Martínez, Gustavo Camarena, Jorge Mercado, y Josefina Martinez. Desarrollo de una BCI utilizando el potencial P300 y la diadema Mindwave®, 2017.
- [13] García Andrade, Adriana, y Adriana García Andrade. «Neurociencia de las emociones: la sociedad vista desde el individuo. Una aproximación a la vinculación sociología-neurociencia». *Sociológica (México)* 34, n.o 96 (abril de 2019): 39-71.

- [14] Biess, Frank y Daniel M. Gross (2014). “Emotional Returns”. En *Science and Emotions after 1945: A Transatlantic Experience*, editado por F. Biess y D. Gross. Chicago: The University of Chicago Press.
- [15] Todo neurociencia. «Qué es la neurociencia y cuáles son sus ramas de estudio», 23 de enero de 2021.
- [16] Koelstra, Sander, Christian Muhl, Mohammad Soleymani, Jong-Seok Lee, Ashkan Yazdani, Touradj Ebrahimi, Thierry Pun, Anton Nijholt, y Ioannis Patras. «DEAP: A Database for Emotion Analysis; Using Physiological Signals». *IEEE Transactions on Affective Computing* 3, n.º 1 (enero de 2012): 18-31.
- [17] Shih, Jerry J., Dean J. Krusienski, y Jonathan R. Wolpaw. «Brain-Computer Interfaces in Medicine». *Mayo Clinic Proceedings* 87, n.º 3 (marzo de 2012): 268-79.
- [18] Tan, Ping, Guanzheng Tan, y Zixing Cai. «Dual-tree complex wavelet transform based feature extraction for brain computer interface». 2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), 2015.
- [19] «EN GRAN MEDIDA... VIBRACIONES – Academia de Ciencias de la Región de Murcia».
- [20] Mihajlović, Vojkan, Bernard Grundlehner, Ruud Vullers, y Julien Penders. «Wearable, Wireless EEG Solutions in Daily Life Applications: What are we Missing? » *IEEE Journal of Biomedical and Health Informatics* 19, n.º 1 (enero de 2015): 6-21.
- [21] Pal, Mayukha, Neeraj, y Prasanta K. Panigrahi. «Coupled Oscillator Dynamics in Brain EEG Signals: Manifestation of synchronization and Across Frequency Energy Exchange by Neutral Turbulence». *IEEE Access* 10 (2022): 20445-54.
- [22] canalSALUD. «¿Qué detecta un electroencefalograma?» *canalSALUD* (blog), 29 de julio de 2016.
- [23] I. Daly, D. Williams, A. Kirke, J. Weaver, A. Malik, F. Hwang, E. Miranda, S.J. Nasuto. «Affective brain-computer music interfacing».
- [24] A. Heraz, R. Razaki, C. Frasson. «Using machine learning to predict learner emotional state from brainwaves».
- [25] Bradley C. Lega, Joshua Jacobs, Michael Kahana. «Human hippocampal theta oscillations and the formation of episodic memories».
- [26] G. Buzsaki. «Rhythms of the brain».



[27] K. Takahashi, “Remarks on emotion recognition from multi-modal bio-potential signals,” in Proceedings of the 2004 IEEE International Conference on Industrial Technology, pp. 1138–1143, Hammamet, Tunisia, December 2004.

[28] X. W. Wang, D. Nie, and B. L. Lu, “EEG-based emotion recognition using frequency domain features and support vector machines,” in in Proceeding of the International Conference on Neural Information Processing, pp. 734–743, Guangzhou, China, 2011.

[29] R.-N. Duan, J.-Y. Zhu, and B.-L. Lu, “Differential entropy feature for EEG-based emotion classification,” in Proceedings of the 2013 6th International IEEE EMBS Conference on Neural Engineering, NER 2013, pp. 81–84, New Jersey, NJ, USA, November 2013.

[30] L. Brown, B. Grundlehner, and J. Penders, “Towards wireless emotional valence detection from EEG,” in Proceedings of the 33rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS 2011, pp. 2188–2191, Boston, MA, USA, September 2011.

[31] N. E. Huang, Z. Shen, S. R. Long et al., “The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis,” in Proceedings of the Proceedings Mathematical Physical and Engineering Sciences, vol. 454, pp. 903–995, 1998.

[32] Sant Pau Centre Terapèutic «ONDAS CEREBRALES, ESTRÉS Y BIOMAGNETISMO EMOCIONAL».

[33] «Entrenar modelos para clasificar datos usando machine learning supervisado - MATLAB - MathWorks España».

[34] «Análisis discriminante lineal». En *Wikipedia, la enciclopedia libre*, 13 de diciembre de 2021.

[35] Sáenz Bajo, N., y M. Álvaro Ballesteros. «Redes neuronales: concepto, aplicaciones y utilidad en medicina». *Atención Primaria* 30, n.o 2 (30 de junio de 2002): 119-20.

Anexos

ANEXO I: Código del proyecto

```
function datos = extraer_caracteristicas_new(tipo_dato,data1, data2, data3, data4, data5, data6, data7, data8 )
```

Figura 41. Función extracción características

```
filtro_entrada=designfilt('bandpassfir','FilterOrder',20,'PassbandFrequency1',3,'PassbandFrequency2',47,'PassbandRipple',1,'SampleRate',128);  
eeg=filter(filtro_entrada, eeg_sf);
```

Figura 42. Código filtro entrada

```
eeg(1,:)=emd(squeeze(eeg(1,:)), 'MaxNumIMF',1);  
eeg(2,:)=emd(squeeze(eeg(2,:)), 'MaxNumIMF',1);  
eeg(3,:)=emd(squeeze(eeg(3,:)), 'MaxNumIMF',1);  
eeg(4,:)=emd(squeeze(eeg(4,:)), 'MaxNumIMF',1);  
eeg(5,:)=emd(squeeze(eeg(5,:)), 'MaxNumIMF',1);  
eeg(6,:)=emd(squeeze(eeg(6,:)), 'MaxNumIMF',1);  
eeg(7,:)=emd(squeeze(eeg(7,:)), 'MaxNumIMF',1);  
eeg(8,:)=emd(squeeze(eeg(8,:)), 'MaxNumIMF',1);
```

Figura 43. Código transformada EMD de las señales

```
for wavesegmentloop=1:29  
    dato1=0;  
    dato2=0;  
    dato3=0;  
    dato4=0;  
    dato5=0;  
    dato6=0;  
    dato7=0;  
    dato8=0;  
    data_aux=zeros(1,framesize);  
    for jj =1:framesize %%se cargan los 2 segundos siguientes+los 1 anteriores de la señal  
        data_aux(1,jj)=eeg(1,start+jj);  
        data_aux(2,jj)=eeg(2,start+jj);  
        data_aux(3,jj)=eeg(3,start+jj);  
        data_aux(4,jj)=eeg(4,start+jj);  
        data_aux(5,jj)=eeg(5,start+jj);  
        data_aux(6,jj)=eeg(6,start+jj);  
        data_aux(7,jj)=eeg(7,start+jj);  
        data_aux(8,jj)=eeg(8,start+jj);  
    end
```

Figura 44. Código inventanado señales

```
start=start+(framesize-1*128);
```



```

for ii =1:framesize
h1=hilbert(data_aux(1,ii));
f1(ii)=angle(h1);
h2=hilbert(data_aux(2,ii));
f2(ii)=angle(h2);
h3=hilbert(data_aux(3,ii));
f3(ii)=angle(h3);
h4=hilbert(data_aux(4,ii));
f4(ii)=angle(h4);
h5=hilbert(data_aux(5,ii));
f5(ii)=angle(h5);
h6=hilbert(data_aux(6,ii));
f6(ii)=angle(h6);
h7=hilbert(data_aux(7,ii));
f7(ii)=angle(h7);
h8=hilbert(data_aux(8,ii));
f8(ii)=angle(h8);
end

for ii =1:framesize-1
dato1=dato1+(1/(framesize-1)*abs(f1(ii+1)-f1(ii)));
dato2=dato2+(1/(framesize-1)*abs(f2(ii+1)-f2(ii)));
dato3=dato3+(1/(framesize-1)*abs(f3(ii+1)-f3(ii)));
dato4=dato4+(1/(framesize-1)*abs(f4(ii+1)-f4(ii)));
dato5=dato5+(1/(framesize-1)*abs(f5(ii+1)-f5(ii)));
dato6=dato6+(1/(framesize-1)*abs(f6(ii+1)-f6(ii)));
dato7=dato7+(1/(framesize-1)*abs(f7(ii+1)-f7(ii)));
dato8=dato8+(1/(framesize-1)*abs(f8(ii+1)-f8(ii)));
end

start=start+(framesize-1*128);

datos_v=[dato1 dato2 dato3 dato4 dato5 dato6 dato7 dato8; datos_v];

```

Figura 45. Código primera diferencia de fase

```

for ii =1:framesize-1
dato1=dato1+(1/(framesize-1)*abs(data_aux(1,ii+1)-data_aux(1,ii)));
dato2=dato2+(1/(framesize-1)*abs(data_aux(1,ii+1)-data_aux(2,ii)));
dato3=dato3+(1/(framesize-1)*abs(data_aux(1,ii+1)-data_aux(3,ii)));
dato4=dato4+(1/(framesize-1)*abs(data_aux(1,ii+1)-data_aux(4,ii)));
dato5=dato5+(1/(framesize-1)*abs(data_aux(1,ii+1)-data_aux(5,ii)));
dato6=dato6+(1/(framesize-1)*abs(data_aux(1,ii+1)-data_aux(6,ii)));
dato7=dato7+(1/(framesize-1)*abs(data_aux(1,ii+1)-data_aux(7,ii)));
dato8=dato8+(1/(framesize-1)*abs(data_aux(1,ii+1)-data_aux(8,ii)));
end
start=start+(framesize-1*128);
datos_v=[dato1 dato2 dato3 dato4 dato5 dato6 dato7 dato8; datos_v];

```

Figura 46. Código primera diferencia de series de tiempo

```

for jj =1:framesize%%señal original
data_aux2(1,jj)=eeg_aux(1,start+jj);
data_aux2(2,jj)=eeg_aux(2,start+jj);
data_aux2(3,jj)=eeg_aux(3,start+jj);
data_aux2(4,jj)=eeg_aux(4,start+jj);
data_aux2(5,jj)=eeg_aux(5,start+jj);
data_aux2(6,jj)=eeg_aux(6,start+jj);
data_aux2(7,jj)=eeg_aux(7,start+jj);
data_aux2(8,jj)=eeg_aux(8,start+jj);
end

```

Figura 47. Código de carga de las señales originales



```
for ii =1:framesize-1
    numerador=numerador+data_aux(1,ii)^2;
    denominador=denominador+data_aux2(1,ii)^2;
    dato1=numerador/denominador;
    numerador2=numerador2+data_aux(2,ii)^2;
    denominador2=denominador2+data_aux2(2,ii)^2;
    dato2=numerador2/denominador2;
    numerador3=numerador3+data_aux(3,ii)^2;
    denominador3=denominador3+data_aux2(3,ii)^2;
    dato3=numerador3/denominador3;
    numerador4=numerador4+data_aux(4,ii)^2;
    denominador4=denominador4+data_aux2(4,ii)^2;
    dato4=numerador4/denominador4;
    numerador5=numerador5+data_aux(5,ii)^2;
    denominador5=denominador5+data_aux2(5,ii)^2;
    dato5=numerador5/denominador5;
    numerador6=numerador6+data_aux(6,ii)^2;
    denominador6=denominador6+data_aux2(6,ii)^2;
    dato6=numerador6/denominador6;
    numerador7=numerador7+data_aux(7,ii)^2;
    denominador7=denominador7+data_aux2(7,ii)^2;
    dato7=numerador7/denominador7;
    numerador8=numerador8+data_aux(8,ii)^2;
    denominador8=denominador8+data_aux2(8,ii)^2;
    dato8=numerador8/denominador8;

end

start=start+(framesize-1*128);

datos_v=[dato1 dato2 dato3 dato4 dato5 dato6 dato7 dato8; datos_v];
```

Figura 48. Código energía normalizada

```
function datos_ocho = ocho_electrodos_caracteristicas(tipo_dato,idParticipant,idVideo)
```

Figura 49. Función de extracción señales DEAP

```
if(idParticipant<10)
    myfilename = sprintf('s0%d.mat', idParticipant);
else
    myfilename = sprintf('s%d.mat', idParticipant);
end
load(myfilename);
```

Figura 50. Código de carga de los ficheros DEAP

```

datastart=128*3;
datalength=8064-128*3;

data1=zeros(1,8064-datastart);

for ii =1:datalength
    data1(1,ii)=data(idVideo,1,ii);
    data2(1,ii)=data(idVideo,17,ii);
    data3(1,ii)=data(idVideo,3,ii);
    data4(1,ii)=data(idVideo,20,ii);
    data5(1,ii)=data(idVideo,11,ii);
    data6(1,ii)=data(idVideo,29,ii);
    data7(1,ii)=data(idVideo,14,ii);
    data8(1,ii)=data(idVideo,32,ii);
end

```

Figura 51. Código de carga de las señales de los 8 electrodos

```

datos_ocho=extraer_caracteristicas_new(tipo_dato, data1, data2, data3, data4, data5, data6, data7, data8);

```

Figura 52. Llamada a la función de extracción de características

```

%%TIPO 1%%
%FELIZ%
idParticipant1 = [1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 4 4
idVideo1 = [18 24 20 40 27 1 16 25 3 19 2 21 26 24 20 12 26 18 8 2 19 14 25 40 9
for i = 1:439
    datos1=ocho_electrodos_caracteristicas(1,idParticipant1(i),idVideo1(i));
    matriz_HVHA=[datos1;matriz_HVHA];
end

% PREOCUPADO%
idParticipant2 =[1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 3 3 3 3 3 3 4 4 4 4 5 5
idVideo2 = [4 31 39 35 15 17 36 34 32 38 37 27 31 38 32 30 33 29 37 21 37 39 35 5
for i = 1:298
    datos2=ocho_electrodos_caracteristicas(1,idParticipant2(i),idVideo2(i));
    matriz_LVHA=[datos2;matriz_LVHA];
end

% TRISTE%
idParticipant3 =[1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3 3 3 3 3
idVideo3 = [13 33 10 14 11 30 29 12 28 23 9 35 28 7 6 23 39 21 22 34 10 25 22 26
for i = 1:278
    datos3=ocho_electrodos_caracteristicas(1,idParticipant3(i),idVideo3(i));
    matriz_LVLA=[datos3;matriz_LVLA];
end

% RELAJADO%
idParticipant4 = [1 1 1 1 2 2 2 2 2 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 4 4
idVideo4 = [5 6 22 7 17 11 4 13 5 36 16 6 32 1 2 8 27 19 14 13 11 12 5 20 40 9 15
for i = 1:262
    datos4=ocho_electrodos_caracteristicas(1,idParticipant4(i),idVideo4(i));
    matriz_HVLA=[datos4;matriz_HVLA];
end

```

Figura 53. Código extracción de matrices de características de las señales de DEAP

```

limit = min(length(matriz_HVHA),min(length(matriz_LVHA),min(length(matriz_LVLA),length(matriz_HVLA))));

matriz_HVHA = matriz_HVHA(1:limit,:);
matriz_LVHA = matriz_LVHA(1:limit,:);
matriz_LVLA = matriz_LVLA(1:limit,:);
matriz_HVLA = matriz_HVLA(1:limit,:);
  
```

Figura 54. Código undersampling matrices de datos

```

matriz_hv= [matriz_HVHA matriz_HVHA2 matriz_HVHA3; matriz_HVLA matriz_HVLA2 | matriz_HVLA3; matriz_LVL
  
```

Figura 55. Código matriz de datos valencia

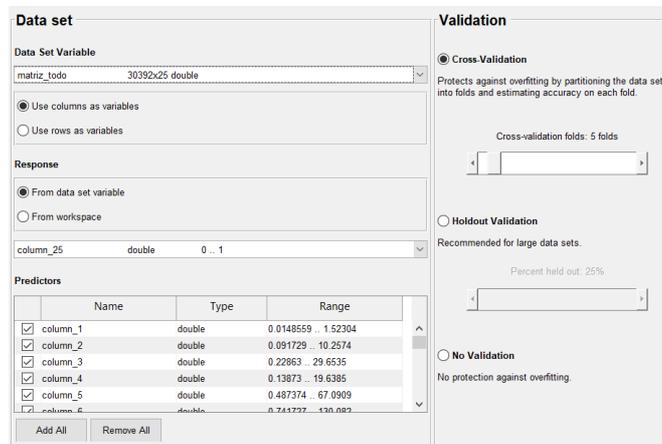
```

m1_out=ones(15196,1);
m2_out=zeros(15196,1);

matriz_out=[m1_out; m2_out];

matriz_todo=[matriz_hv matriz_out];
  
```

Figura 56. Código matriz a introducir en el clasificador



Data set

Data Set Variable: matriz_todo (30392x25 double)

Use columns as variables
 Use rows as variables

Response

From data set variable
 From workspace

column_25 (double) 0 .. 1

Predictors

	Name	Type	Range
<input checked="" type="checkbox"/>	column_1	double	0.0148659 .. 1.52304
<input checked="" type="checkbox"/>	column_2	double	0.091729 .. 10.2574
<input checked="" type="checkbox"/>	column_3	double	0.22863 .. 29.6535
<input checked="" type="checkbox"/>	column_4	double	0.13873 .. 19.6385
<input checked="" type="checkbox"/>	column_5	double	0.487374 .. 67.0909
<input checked="" type="checkbox"/>	column_6	double	0.744292 .. 130.082

Validation

Cross-Validation
 Protects against overfitting by partitioning the data set into folds and estimating accuracy on each fold.

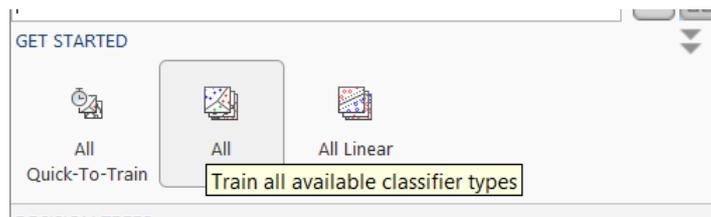
Cross-validation folds: 5 folds

Holdout Validation
 Recommended for large data sets.

Percent held out: 25%

No Validation
 No protection against overfitting.

Figura 57. Menú de la aplicación Classification Learner



GET STARTED

All Quick-To-Train

 All **Train all available classifier types**

 All Linear

Figura 58. Opción de entrenar todos los clasificadores



```
T=matriz_hv(15000:30000,:);  
  
valence = ClasificadorValencia.predictFcn(T) %%clasificador valencia 1 alta 0 baja  
  
T2=matriz_ha(15000:30000,:);  
arousal = ClasificadorExcitacion.predictFcn(T2)
```

Figura 59. Código de uso de los clasificadores

```
if (valence==0)  
    if (arousal==0)  
        tipo_emocion=3;  
    else  
        tipo_emocion=4;  
    end  
else  
    if (arousal==0)  
        tipo_emocion=2;  
    else  
        tipo_emocion=1;  
    end  
end
```

Figura 60. Código especificación del tipo de emoción

```
datos1=ocho_electrodos_caracteristicas(1,5,12);  
datos2=ocho_electrodos_caracteristicas(2,5,12);  
datos3=ocho_electrodos_caracteristicas(3,5,12);  
  
% datos1=extraer_caracteristicas_new(1, data1, data2, data3, data4, data5, data6, data7, data8);  
% datos2=extraer_caracteristicas_new(2, data1, data2, data3, data4, data5, data6, data7, data8);  
% datos3=extraer_caracteristicas_new(3, data1, data2, data3, data4, data5, data6, data7, data8);  
  
datos=[datos1 datos2 datos3];  
  
valence = ClasificadorValencia.predictFcn(datos); %%clasificador valencia 1 alta 0 baja  
arousal = ClasificadorExcitacion.predictFcn(datos);  
  
if (valence==0)  
    if (arousal==0)  
        tipo_emocion=3;  
    else  
        tipo_emocion=4;  
    end  
else  
    if (arousal==0)  
        tipo_emocion=2;  
    else  
        tipo_emocion=1;  
    end  
end
```

Figura 61. Verificación del sistema

```
tipo_emocion =  
  
2
```

Figura 62. Resultado de la verificación

```
function parameters_suspension = fcn(emotion)  
switch emotion  
  
    case 1  
        parameters_suspension = load ('suspension1.mat');  
    case 2  
        parameters_suspension = load ('suspension2.mat');  
    case 3  
        parameters_suspension = load ('suspension3.mat');  
    case 4  
        parameters_suspension = load ('suspension4.mat');  
end  
end
```

Figura 63. Función asignación parámetros de suspensión

ANEXO II: Implementación hardware

Dispositivos

Se ha realizado un estudio del mercado en esta área de aplicación para poder seleccionar la opción más adecuada en este proyecto si se quisiera adquirir un hardware para la implementación final. Aunque hay muchos dispositivos de este tipo, se han descartado en primer lugar todos aquellos de electrodos húmedos y que no sean portátiles puesto que no son compatibles con la aplicación propuesta, es decir, en la conducción.

Dentro del mercado de dispositivos de electrodos secos se han seleccionado las cuatro mejores opciones para este proyecto:

1. Air (Bitbrain): un dispositivo móvil y ergonómico. Realiza una monitorización precisa a través de 8 sensores secos ubicados en las áreas cerebrales pre-frontal y occipital para la estimación básica de estados emocionales y cognitivos. Este incluye Bitbrain Software Kit para la adquisición y programación de datos y tiene alta compatibilidad con plataformas software para el procesamiento de datos real-time como MATLAB, Python, OpenVibe... Funciona a una frecuencia de muestreo 256 Hz a 24 bits.

Electrodos = Fp1, Fp2, AF7, AF8, PO7, PO8, O1, O2

Precio = 6.957,5 €



Figura 64. Dispositivo AIR

2. Nextsem headset (NexStem): es un dispositivo con buena calidad de señal de salida, tiene 16 sensores secos en las zonas frontal y parietal. Tiene conexión bluetooth o wifi, es compatible con MATLAB y su frecuencia de muestreo es de 976.5625 Hz.

Electrodos = AF3, AF4, FZ, C3, C4, C5, C6, CZ, FC3, FC4, CP3, CP4, CPZ, PZ, PO3, PO4

Precio = 841,99 €



Figura 65. Dispositivo NEXSTEM HEADSET

3. DSI 7 (WEARABLE Sensing): indican que es un dispositivo con buena calidad de señal de salida compuesto por 7 sensores secos en posiciones específicas. Puede realizar la transmisión por bluetooth o por cable e incluye el software DSI-Streamer. Su frecuencia de muestreo es de 300Hz a 16 bits.

Electrodos = F3, F4, C3, C4, Pz, P3, P4

Precio = 7.000 €



Figura 66. Dispositivo DSI 7

4. Ultracortex "Mark IV" EEG Headset (OpenBCI): ieste se adquiere en el pack DIY Neurotechnologist's Starter Kit ya que necesita la propia tarjeta OpenBCI para utilizarlo. Consta de 8 electrodos secos ubicados de manera personalizada. También es compatible con MATLAB.

Electrodos = a elegir

Precio = 1.800 €



Figura 67. Dispositivo Ultracortex EEG Headset



Una vez han sido analizadas todas las posibles opciones de hardware, se ha concluido con que la mejor opción para este tipo de aplicación es la cuarta. Esto se debe a que, aunque sea una opción más ruda aparentemente, ya que el proyecto se encuentra todavía en una fase inicial de investigación y desarrollo, es suficiente para hacer validaciones y pruebas antes de adquirir un hardware con mayor coste. Además, esta opción de compra dispone de recursos de las investigaciones realizadas con sus dispositivos, como un foro para los usuarios, además de un apartado de Community para los experimentos realizados. A todo esto, se le suma que tiene un coste no muy elevado y que se puede seleccionar la ubicación de los sensores a nuestra preferencia.