



Universidad
Zaragoza

Proyecto Fin de Carrera

Ingeniería Informática

Proyecto Roujin: Entorno experimental para pruebas de control con señales bioeléctricas

Autor

Carlos Alquézar Baeta

Directores

Manuel González Bedia

Francisco José Serón Arbeloa

Laboratorio ISAAC

Grupo de Informática Gráfica Avanzada

Departamento de Informática e Ingeniería de Sistemas

Escuela de Ingeniería y Arquitectura

Septiembre 2015

A mis padres, por su coraje, su aguante y su fuerza para seguir adelante

A mis tíos, por el cariño y apoyo incondicional que me dieron

A mis mentores, aquí presentes como directores, por lo que me han transmitido

*A todas aquellas personas que en algún momento han hecho posible
que pudiera seguir avanzando*

Porque nunca hay que rendirse...

Gracias

1. Resumen

La rehabilitación de pacientes que han sufrido un accidente cerebrovascular y que arrastran secuelas como parálisis de miembros o hemiplejias, resulta un proceso largo y costoso, donde los fisioterapeutas, a base de ejercicios y entrenamiento, intentan que el paciente “reaprenda” las funciones motoras que ha perdido. Actualmente se está investigando cómo aprovechar el gran avance que se ha producido en los últimos años en el campo de tratamiento de señales bioeléctricas como el electromiograma (EMG, actividad bioeléctrica muscular) o el electroencefalograma (EEG, actividad bioeléctrica del cerebro), para desarrollar novedosas técnicas de rehabilitación, estableciendo nuevos canales de comunicación con el paciente, y ver hasta qué grado esto puede ayudar en la recuperación.

Por esta razón nace el Proyecto Roujin, contexto de este PFC. Este proyecto es una iniciativa estudiantil para el desarrollo de un exoesqueleto orientado a rehabilitación, de bajo coste, controlado por señales bioeléctricas. Este PFC pretende ser una base a medio-largo plazo dentro del Proyecto Roujin, estableciendo las bases conceptuales y metodológicas, y proporcionando un entorno experimental, de bajo coste, en el que se puedan testear e implementar técnicas de control motor a través de señales bioeléctricas. Dicho entorno experimental constará de: (1) un brazo robótico humanoide fabricado con tecnología de impresión 3D; (2) una placa de adquisición de señal EMG; (3) una base de datos de 109 sujetos con más de 1500 registros de EEG; (4) un conjunto de módulos software basados en Matlab que permitirán realizar pruebas de registro, control y evaluación entre las señales bioeléctricas y el brazo robótico.

En este trabajo se trazarán dos líneas de desarrollo. Por un lado, se construirá el brazo robótico, se integrará con la placa de adquisición de señal, se implementarán los módulos de registro y control del sistema, y se procederá a evaluar el control del brazo robótico con la señal electromiográfica del antebrazo derecho de varios sujetos. Por otro lado, se procederá a la evaluación del control del brazo robótico con señal cerebral (EEG) mediante dos técnicas de clasificación: Mapas Auto-Organizados y Análisis Discriminante Lineal. Para ello, se definirán e implementarán las etapas de filtrado de señal, extracción de características y entrenamiento de los clasificadores, y se procederá a la evaluación de los mismos mediante la técnica de validación cruzada aleatoria. Por último se presentará una discusión de éstos resultados.

Índice

1	Resumen	5
2	Introducción	12
2.1	Motivación	15
2.2	Objetivos y Alcance	17
2.3	Herramientas	18
2.4	Organización de la memoria	18
3	Estado del arte	19
4	Esquema general	22
5	Construcción del brazo robótico	24
5.1	ArduinoIO	25
5.2	Servomotores y algoritmo de control	25
5.3	Tabla de tiempos	26
5.4	Alcance	26
6	Control con EMG	27
6.1	Programación de la rutina de interrupción de reloj	28
6.2	Descripción de la placa de adquisición de señal	29
6.3	Algoritmo de clasificación de movimiento	30
6.4	Algoritmo de calibración del umbral	31
6.5	Tabla de tiempos	32
6.6	Alcance	32
7	Control con EEG	33
7.1	Córtex cerebral y EEG.	34
7.2	Base de datos	37
7.3	Preprocesado	39
7.3.1	Análisis de Componentes Independientes	40
7.3.2	Características de los canales defectuosos	40
7.3.3	Características de los artefactos musculares	43

7.3.4	Características de los parpadeos	44
7.4	Extracción de patrones	47
7.5	Métodos de clasificación	48
7.5.1	SOM	48
7.5.2	LDA	49
7.6	Criterios de evaluación	50
7.7	Validación cruzada	51
7.8	Resultados	52
7.8.1	De una única serie	52
7.8.2	De todas las series	55
7.9	Discusión	56
7.10	Tabla de tiempos	58
8	Conclusiones y trabajo futuro	59
	Referencias	61
9	Anexo I: ESQUEMA PLACA EMG	64
10	Anexo II: ANÁLISIS DE COMPONENTES INDEPENDIENTES	65
10.1	Definición de ICA	65
10.2	Ambigüedades de ICA	65
10.3	Independencia vs no correlación	66
10.4	Independencia vs No Gaussianidad	66
10.5	Preprocesado de ICA	67
10.6	Algoritmo FastICA para una unidad	68
10.7	Algoritmo FastICA para varias unidades	69
11	Anexo III: DIAGRAMA DE GANTT	70
12	Anexo IV: TABLA TOTAL DE TIEMPOS	71

Índice de figuras

2.1	Ejemplo de neuro-rehabilitación de miembro superior	13
2.2	Gael Langevin y su InMoov	14
2.3	Algunos proyectos OpenSource	14
2.4	Pipeline del Proyecto Roujin	16
2.5	Líneas de desarrollo del proyecto	17
3.6	Banda Myo, de Thalmic Labs	20
3.7	Esquema del funcionamiento de un sistema BCI	21
4.8	Diagrama de módulos sobre los que se basó el desarrollo	22
4.9	Diagrama de los módulos RP	23
5.10	Pruebas de funcionamiento del brazo robótico	26
6.11	Músculos implicados del antebrazo anterior derecho	27
6.12	Placa de adquisición de señal EMG	29
6.13	Etapas de la placa de adquisición de señal EMG	30
6.14	Método de clasificación con EMG	30
6.15	Pruebas de control con EMG del brazo robótico	32
7.16	Esquema general del sistema de clasificación y sus diferentes etapas.	33
7.17	El córtex cerebral	34
7.18	Montaje usado en la adquisición del EEG	38
7.19	Módulo y submódulos de filtrado de artefactos	39
7.20	Topoplots de la clasificación de canales defectuosos	41
7.21	Boxplots referentes a los coeficientes de correlación entre CIs y canales detectados como defectuosos	42
7.22	Canal Fp1-F3 antes del filtrado de los parpadeos	45
7.23	Canal Fp1-F3 después del filtrado de los parpadeos	46
7.24	Estructura del fichero de registro de EEG.	47
7.25	Capas SOM	48
7.26	Proyección del SOM	49
7.27	Ejemplo representativo de un discriminante lineal de dos clases	50
7.28	Diagramas de caja de las distribuciones de Kappas referentes a una única serie	54
7.29	Diagramas de caja de las distribuciones de porcentajes de acierto referentes a una única serie	54

7.30	Diagramas de caja de las distribuciones de Kappas referentes a todas las series . . .	57
7.31	Diagramas de caja de las distribuciones de porcentajes de acierto referentes a todas las series	57
9.32	Esquema de la placa de adquisición de señal EMG	64
11.33	Diagrama de Gantt del proyecto	70

Índice de cuadros

1	Tabla de tiempos de la fase de construcción del brazo robótico	26
2	Tabla de tiempos de la fase de control con EMG	32
3	Ejemplo de matriz de confusión	51
4	Descriptivo de los Kappas para una única serie	52
5	Descriptivo del caso mejor en la clasificación de una única serie	53
6	Descriptivo del caso peor en la clasificación de una única serie	53
7	Descriptivo de los Kappas para todas las series	55
8	Descriptivo del caso mejor en la clasificación de todas las series	55
9	Descriptivo del caso peor en la clasificación de todas las series	56
10	Tabla de tiempos de la fase de control con EEG	58
11	Tabla total de tiempos del proyecto	71

2. Introducción

El accidente cerebrovascular (también conocido como ictus) es la segunda causa más común de muerte y la principal causa de discapacidad en el mundo [1]. Según dice textualmente la Federación Española de Ictus (FEI), ésta es la primera causa de mortalidad entre las mujeres españolas y la segunda en los varones, según datos del Grupo de Estudio de Enfermedades Cerebrovasculares de la Sociedad Española de Neurología (GEECV-SEN). En Europa mueren 650.000 personas anualmente por esta causa y, de ellas, 40.000 son españolas. Al año se detectan unos 120.000 casos nuevos. De hecho, cada seis minutos se produce un ictus en España. Ocurre cuando se produce una disminución u obstrucción del flujo sanguíneo hacia el cerebro. En un accidente cerebrovascular, un área del cerebro empieza a morir porque deja de recibir el oxígeno y los nutrientes que necesita para funcionar. Las personas que consiguen sobrevivir a un ictus pueden experimentar parálisis o pérdida de fuerza física en un lado del cuerpo (hemiplejía) así como problemas de memoria que dificulten el llevar a cabo actividades de la vida cotidiana. La rehabilitación es el principal tratamiento para estas minusvalías, un proceso que permite a los pacientes reaprender lo mejor posible el uso de sus extremidades y ganar independencia.

La terapia de rehabilitación se puede alargar años, por lo tanto puede ser un proceso largo y laborioso. Se han encontrado ciertos factores que pueden contribuir a recuperar más rápidamente las funciones motoras, aunque todos ellos no se han aprovechado al máximo dada la falta y carencia de servicios de rehabilitación [1]. Se sabe, por ejemplo, que para ciertos casos, la terapia es más efectiva cuanto más pronto se comienza la rehabilitación [2], o que en algunos estudios se ha visto cómo la terapia intensiva y ejercicios basados en resolver tareas contribuyen significativamente a la recuperación de las funciones motoras [3]. La fisioterapia aplicada a estos pacientes que han sufrido un ictus ha mostrado una mejora continuada, sugiriendo que la recuperación puede continuar por muchos años después del accidente cerebrovascular. Como la población de pacientes que sufren este tipo de enfermedades continua creciendo, proveer de un tratamiento adecuado a cada uno de ellos es una tarea cada vez más difícil dado que requiere de una intensa labor especializada.

En este punto, los exoesqueletos orientados a rehabilitación tienen el potencial para hacer frente a la fuerte demanda que la terapia convencional se ve incapaz de asumir. Estas estructuras robóticas son diseñadas para que se adapten a las extremidades y características específicas de cada paciente. Los exoesqueletos tienen el potencial para proveer una rehabilitación intensiva consistente y de larga duración [4], con la ventaja de resolver el problema de la fatiga del terapeuta durante las sesiones. La idea es que estos exoesqueletos puedan llegar a ser utilizados por los pacientes sin la presencia de un especialista, posibilitando que ellos mismos autogestionen la frecuencia de las sesiones de entrenamiento, permitiendo, entre otras ventajas, reducir los altos costes del tratamiento (ver fig. 2.1). Además, otra ventaja de incorporar esta tecnología es que estos robots pueden realizar mediciones precisas y cuantitativas de cómo el paciente está realizando los entrenamientos, proporcionando una información muy valiosa que puede servir tanto para monitorizar como para reforzar aquellos puntos en los que se detecte la necesidad. La combinación de estos exoesqueletos con aplicaciones de realidad virtual diseñadas expresamente

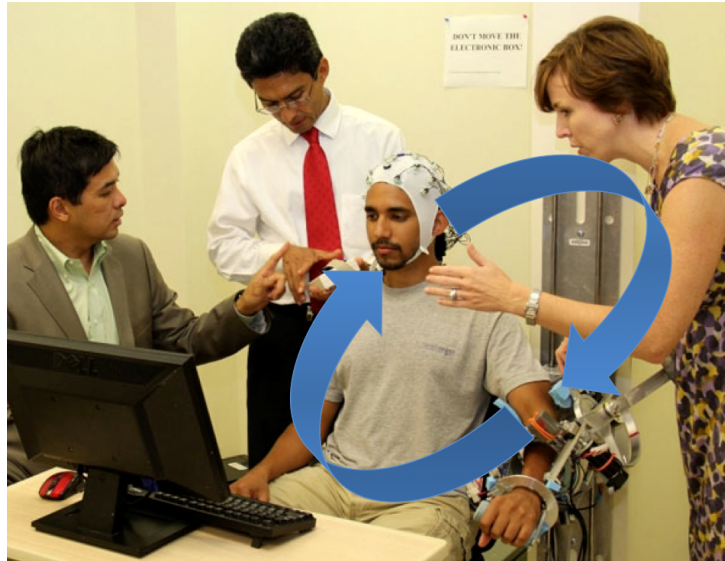


Figura 2.1: Ciclo de neuro-rehabilitación de un miembro superior. El sujeto hace intención de mover el miembro superior, el sistema lo detecta, ejecuta la acción y por feedback visual y propioceptivo el sujeto recibe la información del movimiento de su brazo.

para rehabilitación, pueden ayudar también a mejorar las condiciones en las que el paciente realiza las sesiones de entrenamiento, tanto por el diseño específico de tareas como por hacer de ello una actividad más entretenida y llevadera [5].

Aunque es cierto que en los últimos años se ha experimentado un avance muy significativo tanto en el desarrollo tecnológico como en la reducción de los costes de producción, para las personas con ciertas minusvalías, como pueden ser falta de movilidad o la amputación de miembros inferiores o superiores, todavía les es difícil acceder a equipos de atención y apoyo especializados debido al alto precio al que ascienden. Por ejemplo, para una persona con un miembro superior amputado, un brazo neuroprotésico le puede suponer una inversión de unos 60000 €.

En paralelo, se está produciendo un auge en el uso y distribución de tecnología de prototipado rápido (impresoras 3D), hecho que está causando que emerjan comunidades de lo que se conoce como Makers, en donde éstos comparten conocimiento y diseños, y en donde se aprovechan de la experiencia de otros proyectos para desarrollar los suyos propios. Estas comunidades, beneficiándose de la alta conectividad que ofrece ahora mismo internet, adoptan una estructura distribuida y libre en donde el intercambio de recursos es uno de los pilares esenciales sobre los que se asientan.



Figura 2.2: El autor del robot InMoov, Gael Langevin, con su creación.

Dentro de estas comunidades, destacan ciertos grupos de trabajo que se están especializando en intentar proporcionar herramientas de bajo coste para suplir el problema de acceso a algo tan importante como puede ser una neuroprótesis o un brazo robótico para alguien que carece de algún miembro. En esta línea, proyectos como [InMoov](#), [RoboHand](#) o [Project Daniel](#) (ver figuras 2.2 y 2.3) han nacido dentro de estas comunidades de Makers para hacer de esta idea una realidad y llevar a aquellos que lo necesiten esta tecnología. Si bien es cierto que las prestaciones de éstas herramientas no son comparables con las funcionalidades y robustez que ofrecen los símiles comerciales, se han conseguido desarrollar prótesis de miembro superior disminuyendo considerablemente los costes, demostrando su utilidad práctica en personas amputadas.



(a) InMoov

(b) Liam, con su RoboHand

(c) Project Daniel

Figura 2.3: Algunos de los proyectos OpenSource actuales que están empleando la tecnología de prototipado rápido para proporcionar herramientas de bajo coste.

2.1. Motivación

Aunque es cierto que esta tecnología está en creciente desarrollo y las comunidades que emergen cada vez son más numerosas, la mayoría de proyectos que surgen están orientados al desarrollo de neuroprótesis para personas que han perdido alguno de sus miembros.

Después de un tiempo de exploración, y de recibir ayuda y recomendaciones de diferentes grupos de investigación como el Brain-Machine Interface Systems Lab (<http://bmi.umh.es>), de la Universidad Miguel Hernández de Elche, y con el apoyo tanto del Grupo de Informática Gráfica Avanzada (<http://giga.cps.unizar.es>) como del Grupo de Robótica, Visión y Tiempo Real (<http://robots.unizar.es>) de la Universidad de Zaragoza, nace el Proyecto Roujin. Este proyecto es una iniciativa de dos estudiantes de la Universidad de Zaragoza que, si bien se estaban llevando a cabo diferentes proyectos orientados a neuroprótesis, encontraron que no existían equivalentes en temas de rehabilitación. Decidieron entonces poner en marcha el Proyecto Roujin con el objetivo de desarrollar un exoesqueleto orientado a rehabilitación de bajo coste, para así proporcionar una herramienta de trabajo asequible a docentes e investigadores y permitir el acceso a esta tecnología a cuantas más universidades y centros de investigación sea posible.

Dada la envergadura de esta iniciativa, la escasez de recursos humanos, técnicos y económicos disponibles, y que nuestra formación era complementaria pero distinta (yo aportaba un perfil informático y mi compañero un perfil industrial) se decidió, en una primera instancia, segmentar el desarrollo en tres bloques temáticos, de los cuales cada uno abordaría uno de ellos en sendos PFC's y dejando un tercero para un desarrollo posterior. Estos bloques fueron:

- Tratamiento de señales bioeléctricas.
- Programación del control orientado a rehabilitación.
- Diseño del exoesqueleto.

El orden natural de estos bloques se puede ver en la figura 2.4, donde el bloque de tratamiento de señal es el encargado de procesar la información de las señales electrofisiológicas del sujeto, el bloque de control hace uso de esta información preprocesada para determinar las acciones que debe tomar, y finalmente el exoesqueleto ejecuta las ordenes.

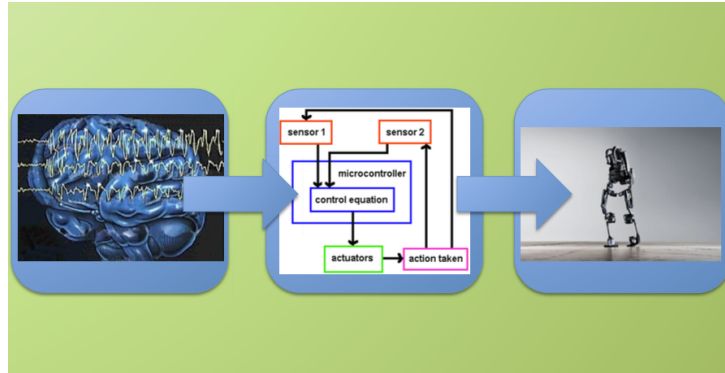


Figura 2.4: Pipeline del Proyecto Roujin

Sabiendo que éste es un tema actual de investigación al que se dedican grandes grupos de diferentes centros y laboratorios, y que además cuentan con financiación pública y privada, el objetivo que nos planteamos fue el de que con nuestros proyectos final de carrera se establecieran unas primeras bases en el desarrollo general, adquirir conocimiento y experiencia en el tema, e intentar darle difusión para que otros estudiantes puedan tomar el relevo en un futuro. Uno de los fundamentos del Proyecto Roujin es el de que éste pueda servir para impartir docencia y dar la oportunidad a estudiantes para formarse de manera práctica en estos temas. Creemos que en una escuela de ingeniería en donde coexisten diferentes disciplinas, es necesario, no solo mostrarle al estudiante los últimos avances llevados a cabo en las diferentes ramas de su profesión, sino también habilitar espacios y proyectos comunes en donde diferentes perfiles puedan converger para así enriquecer su formación y favorecer sinergias. Creemos que éste podría ser uno de ellos. El Proyecto Roujin no está pensado para ofrecer una alternativa a productos equivalentes que ofrece la industria, sino que dada su filosofía OpenSource, pretende ser una herramienta gracias a la cual más gente pueda formarse en temas de neuro-rehabilitación. Una de las formas en las que se puede mejorar esta línea de investigación, es aumentar el orden de magnitud de personas formadas en el tema, repercutiendo así en la competencia y facilidad para encontrar mejores profesionales.

El otro proyecto final de carrera asociado a éste llevaba por título *“Proyecto Roujin: Diseño de un exobrazo robótico de 4 grados de libertad orientado a rehabilitación por impresión 3D”* y fue presentado el pasado Febrero de 2015 por Adrián Blasco Carrillo.

2.2. Objetivos y Alcance

Como ya se ha dicho anteriormente, en este proyecto final de carrera se ha pretendido abrir camino en una línea de trabajo orientada al control de neuroprótesis y exoesqueletos para rehabilitación motora, mediante el procesamiento de señales electrofisiológicas. Por ello, este PFC pretende ser una base a medio-largo plazo dentro del Proyecto Roujin.

Cuatro objetivos fueron definidos: (1) construcción de brazo robótico para validar y testear algoritmos de control; (2) implementación de un sistema capaz de detectar mediante señal electromiográfica (EMG) del antebrazo, si la mano del sujeto estaba abierta o cerrada, y controlar con ello el brazo robótico; (3) Evaluar la posibilidad de controlar la apertura y cierre de la mano del brazo robótico mediante análisis de señal electroencefalográfica (EEG) de varios sujetos; (4) Autofinanciar el proyecto.

Estos objetivos fueron definidos así porque, con su resolución, se estableció una base conceptual y metodológica para abordar tanto problemas relacionados con la mecánica y electrónica, como con el procesamiento de señales electrofisiológicas.

En la figura 2.5 se puede ver un diagrama con las dos líneas de desarrollo de las que consta este proyecto. Además de la construcción del brazo, para poder llevar a cabo el resto de objetivos, se implementaron una serie de módulos apoyados en Arduino y Matlab, llamados RP MODULES, que forman la base estructural del entorno experimental y sobre los que se apoyarán futuros desarrollos y experimentos.

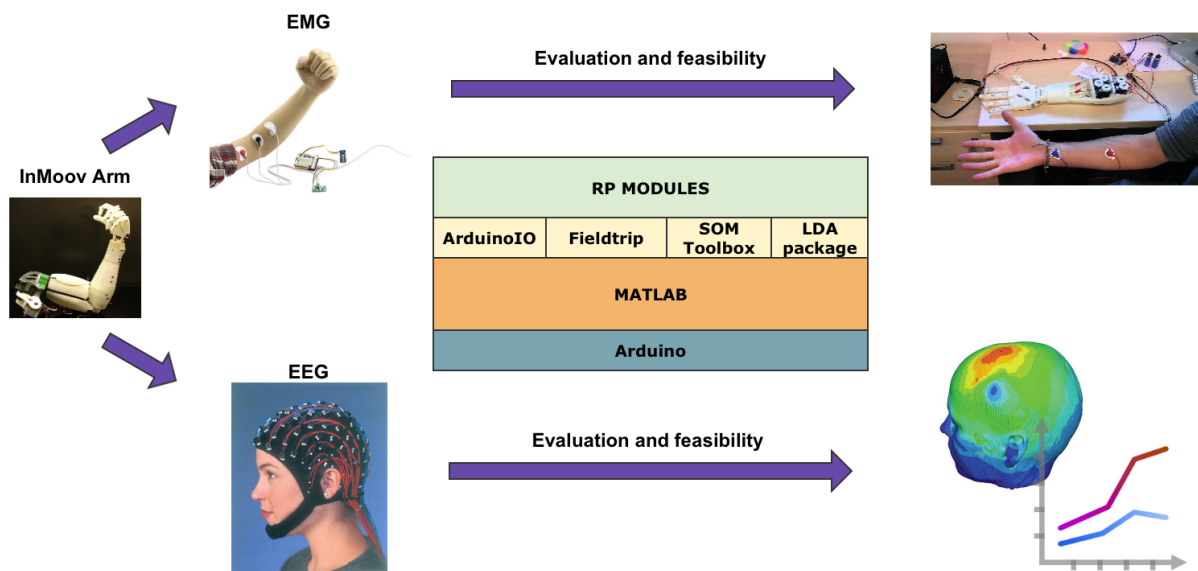


Figura 2.5: Líneas de desarrollo del proyecto

2.3. Herramientas

Para la construcción del brazo robótico, además de fungibles y herramientas comunes, fue necesaria una impresora 3D para la impresión de las piezas de plástico. También fueron necesarios 5 servomotores TowardPro modelo MG996R y una fuente de alimentación de PC. La placa que actuó de controlador fue un Arduino UNO. Para la adquisición de señal EMG, se compró la placa Muscle v3 Sensor de Advancer Technologies. Como plataformas software se utilizaron ArduinoIDE y Matlab. Las toolboxes de Matlab utilizadas fueron ArduinoIO para la comunicación con Arduino, y Fieldtrip, SOM Toolbox y LDA Package para el procesamiento de señal cerebral.

2.4. Organización de la memoria

Dado que este proyecto final de carrera está dividido en tres bloques claramente diferenciables, se ha optado por exponer cada uno de ellos en secciones distintas, dejando una sección previa a modo de esquema general que articula el resto de la memoria. De esta forma se favorece la legibilidad. Al final del documento se encuentra una serie de anexos como complemento a la memoria.

3. Estado del arte

Aunque esta sección se titule Estado del Arte, dadas las restricciones de extensión a las que se ha de ceñir este trabajo, y dado que éste es un proyecto de integración en el que confluyen varios campos, lejos de tomar el significado que en la comunidad científica adquiere, aquí se va a tratar de comentar brevemente el panorama actual en cada uno de estos campos, ofreciendo algunos punteros para que el lector profundice más en el tema en caso de interés.

Mundo Opensource

En los últimos años se ha producido un enorme auge en lo que se refiere a la filosofía DIY (“Do It Yourself”, que en español viene a ser “Háztelo tu mismo”). Se han formado comunidades (a cuyos integrantes se les suele llamar “Makers”) en donde, aprovechando la alta conectividad que ofrece internet, se comparten diseños, modelos, consejos,... en general, conocimiento. La idea principal que subyace en que, si todos comparten, todos estarán más cerca de llegar a sus objetivos personales. Existen para ello portales como Thingiverse (<https://www.thingiverse.com>), centrado en diseños CAD para impresión 3D, en donde los usuarios pueden colgar sus trabajos para que cualquiera se los pueda descargar y adaptar a sus necesidades. Es por ello que muchos proyectos han surgido como “Spin-offs” de otros, tomando como base los diseños de otro autor y realizando los cambios oportunos. Un ejemplo de estas comunidades podría ser E-NABLE (<http://enablingthefuture.org>) centrada en el desarrollo de prótesis de mano por impresión 3D.

Dentro de estas comunidades se pueden encontrar proyectos de todo tipo, desde simples pasatiempos, como pudo ser en sus inicios el robot InMoov (<http://www.inmoov.fr>), hasta proyectos con un fin social concreto como pueden ser proyectos relacionados con la salud y el bienestar. Ejemplos son RoboHand (<http://www.robohand.net>), Project Daniel (<http://www.notimpossible.com/labs/project-daniel>) u OpenBionics (<http://www.openbionics.org>), que se centran en el diseño y desarrollo de neuroprótesis de miembro superior, reduciendo su coste entre 30 y 40 veces lo que cuesta un equivalente comercial en el mercado.

Control con EMG

Los sistemas de control mioeléctricos han sido usados durante muchos años para controlar dispositivos tanto asistenciales como de rehabilitación, mediante la clasificación de patrones en la señal electromiográfica registrada en la superficie de la piel [6] [7]. Mucha de la literatura relacionada con el reconocimiento de patrones en EMG está centrada en la optimización de los clasificadores y en el número de movimientos que pueden ser distinguidos [6] [8]. Aunque la precisión a la hora de clasificar movimientos ha alcanzado valores superiores al 90 % de acierto para la discriminación de múltiples movimientos, haciendo uso de varias combinaciones de avanzadas técnicas de extracción de características, reducción de la dimensionalidad y clasificación de patrones [7], la usabilidad de estos sistemas todavía está limitada por varias cuestiones [9] [10] [11].

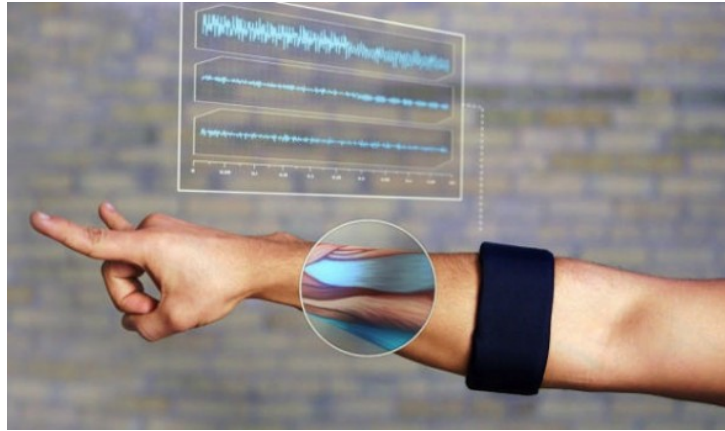


Figura 3.6: Banda Myo, de Thalmic Labs

Aun así, están surgiendo en el mercado dispositivos que hacen uso de la actividad muscular para fines más generales y para todo tipo de público. Por ejemplo, el Myo Gesture Control (ver fig. 3.6), de Thalmic Labs (<https://www.myo.com>), ofrece una API de desarrollo para que las empresas puedan conectar su software con dicho dispositivo y poder manejar mediante gestos tareas comunes como pueden ser el paso de diapositivas en una presentación PowerPoint, el panel de control de reproductores multimedia, o incluso conectarlo con las Google Glass.

Control con EEG

Los interfaces Cerebro-Ordenador son una tecnología emergente que intenta conectar nuestro cerebro natural con dispositivos artificiales, estableciendo un nuevo canal de comunicación entre las señales eléctricas provenientes de la actividad cerebral y dichos dispositivos, sin utilizar los mecanismos neuromusculares naturales [12] [13]. Un interfaz Cerebro-Ordenados (o “BCI” por sus siglas en inglés), reconoce la intención del usuario a través de su actividad cerebral, decodifica la información proveniente de dicha actividad, y la traduce en comandos de salida relacionados con el objetivo de la acción que esté llevando a cabo el usuario. Estos sistemas se pueden clasificar principalmente en base a dos criterios: su grado de invasión corporal, y el tipo de actividad cerebral que procesa.

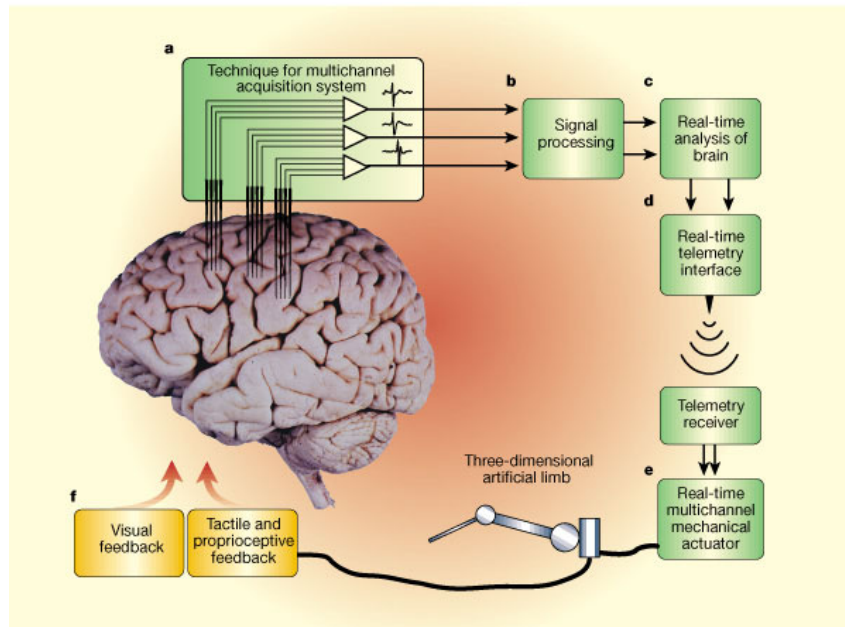


Figura 3.7: Esquema del funcionamiento de un sistema BCI

Respecto al primer criterio, podemos diferenciar dos tipos: sistemas invasivos y sistemas no invasivos. Los sistemas invasivos son aquellos en los que al paciente se le aplica cirugía para introducirle los electrodos en el cerebro a través del cráneo. Esto tiene la ventaja de que aumenta considerablemente la calidad de la señal adquirida, pero tiene el inconveniente del riesgo que asume el paciente. Los sistemas no invasivos son aquellos en los que no se aplica cirugía, y la actividad cerebral se registra en la superficie del cuero cabelludo. El EEG es un método no invasivo.

Respecto al segundo criterio, tenemos, por ejemplo, sistemas basados en potenciales evocados (P300, SSVEP,...), sistemas basados en neurofeedback para la autorregulación de ritmos cerebrales, o sistemas que mediante cierto tipo de ritmos, permiten al sujeto controlar un dispositivo como puede ser un brazo robótico.

El proceso sería el siguiente (ver figura 3.7): (a) mediante unos electrodos colocados en la superficie del cuero cabelludo se registra el EEG de las zonas de interés; (b) las señales EEG son procesadas para mejorar la calidad de las mismas aplicando cierto tipo de filtros y técnicas para ello; (c) una vez la señal ha sido procesada, se procede a la extracción de características para identificar la orden del sujeto; (d) una vez identificada dicha orden, es enviada al dispositivo; (e) el dispositivo recibe la orden y la ejecuta; (f) el sujeto recibe feedback visual o propioceptivo de si el dispositivo ha ejecutado la orden correcta.

4. Esquema general

En este proyecto se desarrollaron una serie de módulos principalmente en Matlab para la integración tanto del brazo robótico como las implementaciones de los respectivos controles por EMG y EEG. En la figura 4.8 se puede ver un diagrama en donde se muestran los bloques en los que se apoyan estos módulos. En lo que a programación se refiere, la principal carga de desarrollo consistió en la programación de los RP MODULES (RP son las siglas de “Roujin Project”).

Durante la fase de diseño se decidió que los módulos implementados ofrecieran diferentes índices de granularidad, relativa a su función, con el objetivo de favorecer la escalabilidad y flexibilidad para futuras modificaciones. Esto desembocó en módulos con alta generalización los cuales debían hacer uso de módulos con alta especialización para desempeñar sus funciones. La razón fue porque este PFC pretende ser un punto de partida para futuros trabajos, y como el control de neuroprótesis y exoesqueletos es un campo todavía en investigación, era necesario habilitar el sistema para poder testear en un futuro nuevos métodos dentro de este paradigma.

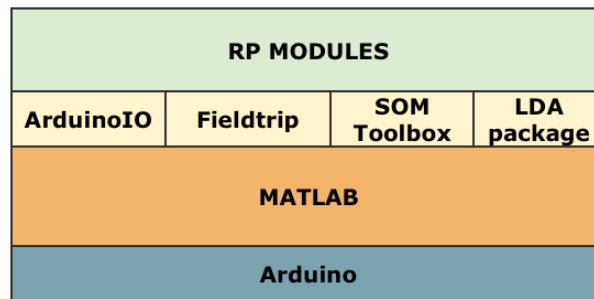


Figura 4.8: Diagrama de módulos sobre los que se basó el desarrollo

En la figura 4.9 se puede ver un diagrama con los módulos implementados en Matlab. En color verde, se muestran aquellos módulos con alta especialización y que podrían ser sustituidos por otros módulos equivalentes. En su caso, con una mínima reconfiguración del resto de módulos el sistema quedaría adaptado.

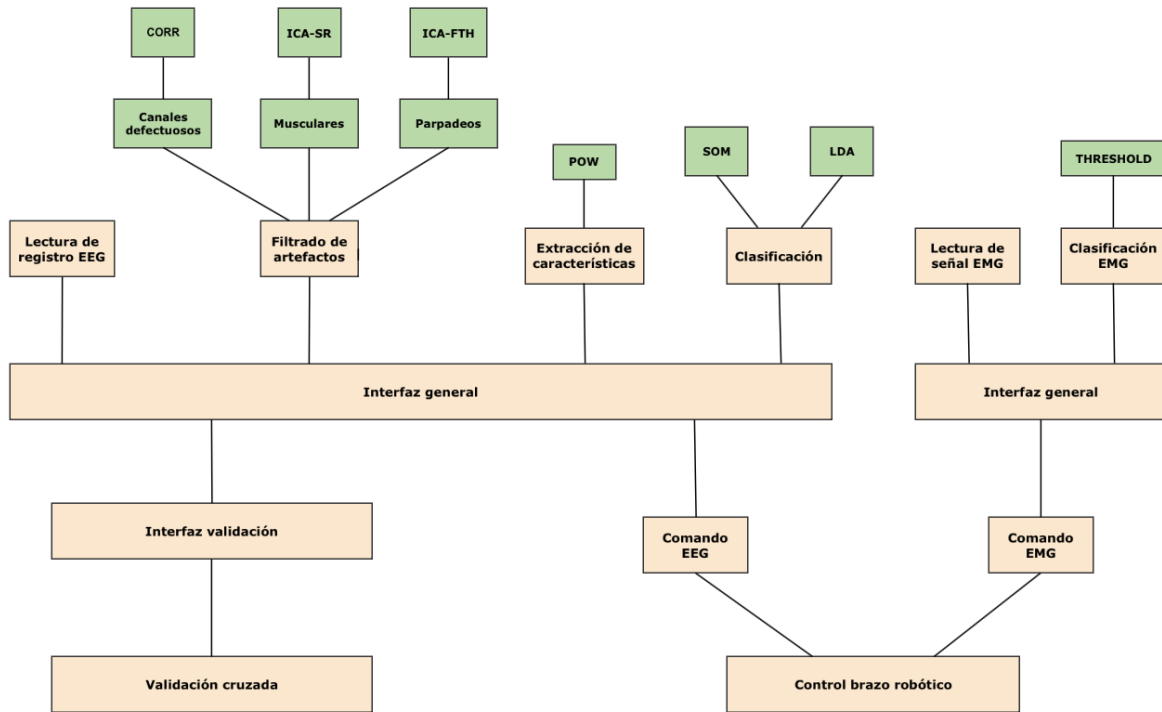


Figura 4.9: Diagrama de los módulos RP

5. Construcción del brazo robótico

En esta fase del proyecto se llevó a cabo la construcción e integración de una herramienta con la que poder realizar pruebas de control. Se decidió construir un brazo robótico humanoide por impresión 3D por varias razones: (1) su bajo coste de producción; (2) utilizar esta tecnología permitiría ganar experiencia y conocimiento sobre la misma para futuros trabajos; (3) un modelo humanoide permite plantear cuestiones de ergonomía y usabilidad; (4) la repercusión mediática es mucho mayor que con otro tipo de modelos; (5) un modelo de estas características se puede reutilizar para posteriores experimentos fuera del ámbito de este proyecto, más próximos a la psicología y neurociencia.

Entre las opciones disponibles, se optó por reutilizar los diseños del robot InMoov (www.inmoov.fr) ya que éstos están disponibles bajo licencia Creative Commons en la plataforma Thingiverse (www.thingiverse.com). Se decidió construir la mano derecha, incluido el antebrazo, hasta el codo. Para este proyecto se consideró que con los 5 grados de libertad de esta parte del brazo (uno por dedo de la mano), era más que suficiente, dado que los clasificadores a implementar solo iban a contemplar dos estados: mano abierta y mano cerrada.

Objetivos

Los objetivos de esta fase del proyecto fueron, además de la construcción propiamente dicha del brazo robótico, implementar un control del brazo tanto en lenguaje nativo de Arduino, como en Matlab. Estas implementaciones servirían como base para los posteriores desarrollos del control utilizando señales electrofisiológicas.

Retos/Dificultades encontradas

Durante esta fase, hubo que repasar los conocimientos de electricidad, tanto prácticos como de instrumentación, sobre todo para reorganizar las líneas de alimentación, y minimizar la complejidad del montaje. También hubo que aprender a programar scripts en lenguaje nativo de Arduino, cargar scripts en el controlador, y depurar errores con ArduinoIDE. Para el control de los servomotores, hubo que repasar conocimientos de actuadores y autómatas programables. También, tanto para comunicar el brazo robótico con Matlab, como para monitorizar el funcionamiento del mismo, hubo que emplear nociones de transmisiones digitales como el baudaje o comunicación serie. Por último, en relación con la comunicación entre Arduino y Matlab, se utilizó un paquete de interfaz llamado ArduinoIO, que permite abstraerse de la comunicación y mandar comandos a Arduino directamente con instrucciones de Matlab. Este interfaz requirió un trabajo de instalación y comprensión para poder ser utilizado en este proyecto.

5.1. ArduinoIO

Este paquete de software permite que desde Matlab se puedan enviar comandos a Arduino como si en lenguaje nativo se tratase, de forma transparente. Para ello, primero hay que instalar un script en Arduino que dependerá del modelo de la placa y si vamos a usar o no el *ADAFRUIT motor shield* (hardware para el control y testeo de motores). Este script funciona a modo de servidor en Arduino y continuamente se está ejecutando a la espera de recibir comandos a través del puerto serie.

Por otra parte, desde Matlab también hay que añadir a nuestro PATH una serie de ficheros con las funciones que nos abstraen de la parte de comunicación y que nos permiten encapsular comandos propios de Arduino en funciones de Matlab.

5.2. Servomotores y algoritmo de control

Para este montaje se utilizaron 5 Servos TowardPro modelo MG996R, controlados por PWM (Pulse-width modulation). Cada servo está ligado con un único dedo de la mano mediante un hilo (a modo de tendón). Este hilo está enlazado con un disco que rota con el giro del servo, y dicha rotación puede ir de 0° a 180° aproximadamente. El diseño se realizó de tal manera que si el servo se disponía en uno de los límites de rotación, el hilo quedaría totalmente extendido, y así como fuese rotando el hilo iría tirando del dedo hasta llegar a la flexión total cuando se alcanzase el otro extremo de rotación. Esto es común a todos los servos pero en el montaje se vio que no todos compartirían los mismos límites para definir flexión y extensión, por lo que en algunos casos, los dedos quedarían extendidos cuando sus servos se posicionaran a 0° , y en otros casos a 180° , siendo al revés para el caso de la flexión. Esto hubo que hacerlo de forma manual, analizando que montaje ofrecía mejores beneficios en términos de fricción, que no se estorbaran unos servos con otros, etc.

Algoritmo de control de los servos

```
% Definiciones iniciales
- A cada servo, se le asigna un pin PWM del Arduino
- Para cada servo, se definen las posiciones de dedo abierto y dedo cerrado

% Dentro del bucle principal
Si (condición X = mano abierta) entonces
    Asignar a cada servo los valores correspondientes con mano abierta
sino
    Asignar a cada servo los valores correspondientes con mano cerrada
fin Si
```

5.3. Tabla de tiempos

Las tareas y tiempos referentes al desarrollo de esta fase del proyecto fueron:

<i>Tarea</i>	<i>Horas</i>
Impresión de las piezas	20
Ensamblaje	15
Conexión con Arduino	16
Programación del control	10
Total	61

Cuadro 1: Tabla de tiempos de la fase de construcción del brazo robótico

5.4. Alcance

Los objetivos planteados en esta fase del proyecto se superaron satisfactoriamente. Se programó un script de pruebas en el que, durante 2 minutos, cada 20 segundos el brazo cambiaba de posición entre mano abierta y mano cerrada. Se obtuvo un 100 % de eficacia.

En este vídeo de youtube <https://youtu.be/nJrS40FLcIO> se puede ver una demo del brazo en funcionamiento.

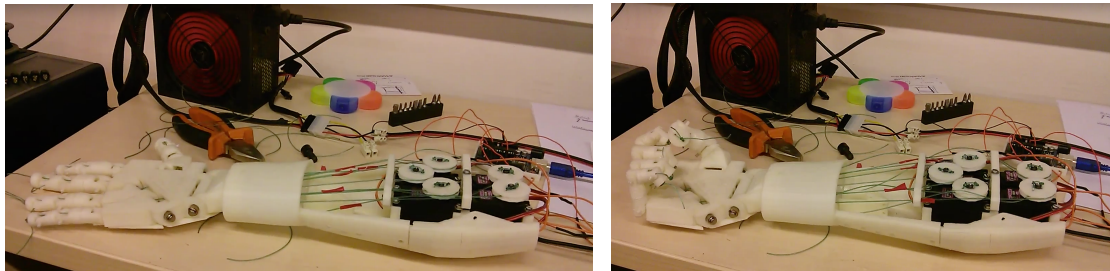


Figura 5.10: Pruebas de funcionamiento del brazo robótico

6. Control con EMG

Esta fase del proyecto consistió en implementar un control del brazo robótico con señal electromiográfica del antebrazo. Dicha señal refleja la actividad eléctrica de los músculos esqueléticos y contiene información acerca de la función de control que sobre los músculos tienen, tanto el sistema nervioso central, como el periférico. La señal EMG ofrece una caracterización muy valiosa del sistema neuromuscular dado que muchos procesos patológicos, tanto si aparecen en el sistema nervioso como en los músculos, se manifiestan por cambios en las propiedades de la señal.

La señal EMG se puede registrar tanto de forma invasiva, aplicando cirugía, como de forma no invasiva, colocando electrodos en la superficie de la piel. La ventaja de los métodos invasivos es su precisión a la hora de registrar actividad eléctrica de un músculo en concreto, o incluso a una escala menor, de ciertas zonas de dicho músculo. La ventaja de los métodos no invasivos es que no hay que realizar ningún tipo de operación, pero tienen la desventaja de que se pierde resolución en el registro, y la señal obtenida se corresponde con la superposición de las señales procedentes de todos aquellos músculos próximos a la zona donde se han colocado los electrodos. Para lecturas grosso modo, este método es suficiente. Se decidió que se aplicaría un método no invasivo para el registro de EMG.

Si bien es cierto que para realizar el control del brazo robótico hubiese sido suficiente el registro de cualquier músculo que el sujeto fuera capaz de contraer y expandir de forma voluntaria, se optó por una configuración realista y se decidió aplicar el registro a los músculos del antebrazo anterior derecho. En concreto, los músculos principalmente implicados son el músculo flexor superficial de los dedos, el flexor profundo de los dedos y el flexor largo del pulgar (vease fig. 6.11).

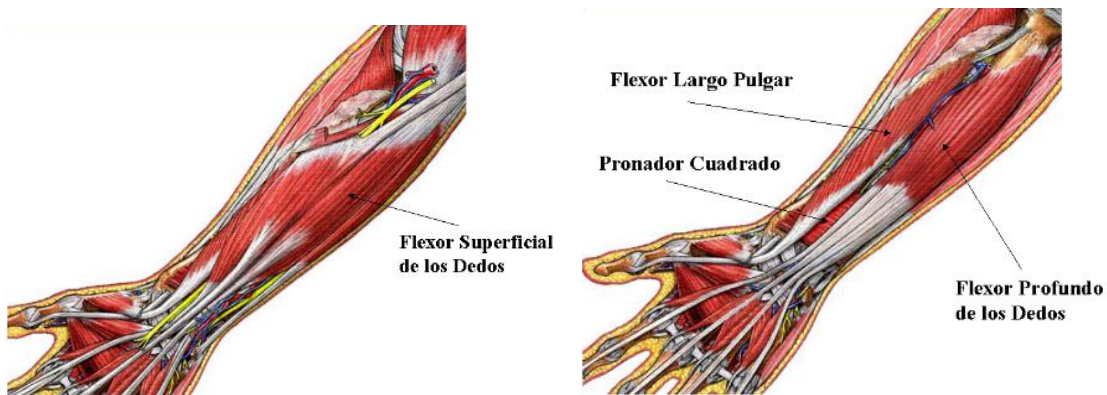


Figura 6.11: Músculos implicados del antebrazo anterior derecho.

Objetivos

Los objetivos que se marcaron para esta fase fueron: (1) inspeccionar el mercado en busca de una placa de adquisición de señal EMG asequible y que se pudiese incorporar de forma fácil al sistema montado en la fase anterior; (2) Integrar la placa de adquisición con Arduino y Matlab; (3) Implementar una interrupción de reloj en Arduino para poder realizar un muestreo regular de la señal; (4) Implementar un control del brazo robótico en Matlab clasificando dos estados (mano abierta/cerrada) mediante la señal EMG adquirida del antebrazo anterior derecho.

Retos/Dificultades encontradas

Durante esta fase, hubo que aplicar de nuevo nociones de electricidad y soldadura para integrar la placa de adquisición de señal al sistema. También hubo que reparar nociones de electrónica analógica como, por ejemplo, los amplificadores operacionales, para poder comprender las especificaciones de la placa de adquisición de señal. Dado que en el plan de estudios de Ingeniería informática no existen asignaturas de neuroanatomía o electrofisiología, hubo que invertir tiempo en formarse en este campo. La programación de la rutina de interrupción de reloj para obtener un muestreo regular obligó a adquirir nociones de sistemas empujados y electrónica digital. Por último, además de implementar el algoritmo de control, dado que las características electrofisiológicas de los músculos son parecidas pero no iguales entre las personas, hubo que diseñar un algoritmo de calibración para que el sistema se adaptase a cada usuario y pudiese funcionar de forma óptima.

6.1. Programación de la rutina de interrupción de reloj

Arduino UNO tiene tres timers (Timer0, Timer1 y Timer2). Cada uno de ellos tiene un contador que se incrementa en cada tick de reloj. El reloj del Arduino UNO funciona a una frecuencia de 16MHz.

Las interrupciones de reloj CTC (Clear Time on Compare) se disparan cuando el contador de uno de los timers alcanza un valor preestablecido y almacenado en el registro de comparación (Compare Match Register). Cuando el timer alcanza el valor especificado, se resetea a 0 en el siguiente tick de reloj, y vuelve a incrementarse de nuevo. Eligiendo el valor de comparación adecuado y la velocidad a la que se incrementa el contador, se puede controlar la frecuencia de las interrupciones. Los contadores de Timer0 y Timer2 son de 8 bits, mientras que el contador de Timer1 es de 16 bits.

El parámetro prescaler permite controlar la frecuencia de incremento de los contadores. Los valores que puede tomar este parámetro son: 1, 8, 64, 256 y 1024:

$$\text{Timer Speed (Hz)} = \frac{\text{Arduino Clock Speed (16MHz)}}{\text{Prescaler}} \quad (1)$$

Para calcular la frecuencia de interrupción, se puede hacer con la siguiente fórmula:

$$\text{Interrupt Frequency (Hz)} = \frac{\text{Arduino Clock Speed (16MHz)}}{\text{Prescaler} \times \text{Compare Match Register} + 1} \quad (2)$$

Para saber qué valor he de tener en el registro de comparación, para una frecuencia de interrupción concreta, dada la fórmula anterior, despejamos y nos queda:

$$\text{Compare Match Register} = \left[\frac{\text{Arduino Clock Speed (16MHz)}}{\text{Prescaler} \times \text{Desired Interrupt Frequency (Hz)}} \right] - 1 \quad (3)$$

Los comandos que se quieran ejecutar después de la interrupción de reloj, han de ir en una rutina diferente a las de `setup()` y `loop()`, cuyo nombre ha de ser:

`ISR(TIMER0_COMPA_vect){}`

Como su nombre indica, el prefijo `TIMER0` se corresponde con el `timer0`, y se deberá cambiar a `TIMER1` cuando la interrupción se desee invocar a través del `timer1`, o a `TIMER2` cuando se desee utilizar el `timer2`.

6.2. Descripción de la placa de adquisición de señal

La placa por la que se optó, que se puede ver en la fig. 6.12, fue la `Muscle Sensor v3` de `Advancer Technologies` (ver esquema en Anexo I [9]). Esta decisión se tomó dada su simplicidad y facilidad de integración con arduino. Consta de un único canal de adquisición.

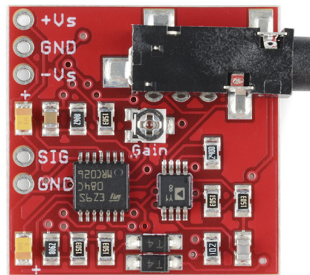


Figura 6.12: Placa de adquisición de señal EMG

Esta placa, además de la etapa de amplificación de señal, tiene dos etapas adicionales de rectificación y suavizado de la señal. En la fig. 6.13 se puede ver, utilizando de referencia una señal sinusoidal a modo de ejemplo, el efecto de las etapas de rectificación (figura del medio) y suavizado (figura de abajo).

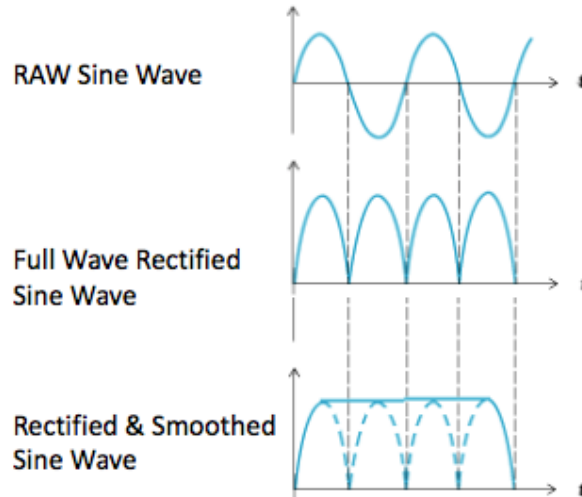


Figura 6.13: Etapas de la placa de adquisición de señal EMG

6.3. Algoritmo de clasificación de movimiento

Se planteó diferenciar cuando el sujeto tuviese la mano abierta y la mano cerrada. El registro de la actividad electromiográfica se realiza colocando un electrodo a mitad del músculo de interés, y otro electrodo al final de éste. Existe una correlación entre la tensión muscular provocada voluntariamente por el sujeto, y la amplitud de la señal electromiográfica registrada.

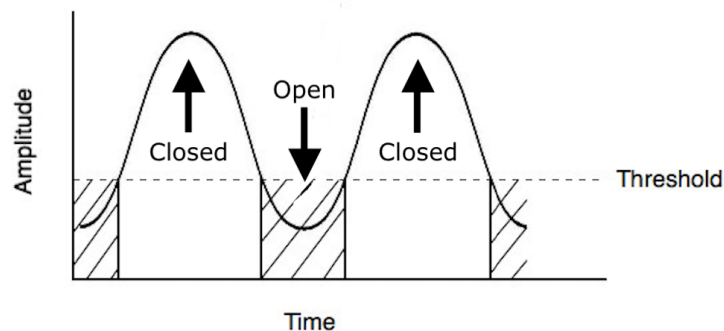


Figura 6.14: El umbral delimita cuando esta abierta o cerrada la mano.

El algoritmo que se planteó para esta primera clasificación fue, aprovechando la relación anterior, situar un umbral a un determinado voltaje, que determinase cuándo se consideraba que el músculo estaba flexionado o no (ver fig. 6.14). Teniendo en cuenta dicha relación, si la

señal sobrepasaba el umbral, se consideraba que el músculo estaba flexionado (mano cerrada), y si la señal se encontraba por debajo, el músculo estaba extendido (mano abierta).

6.4. Algoritmo de calibración del umbral

Dado que las características del brazo derecho de cada sujeto variaban, esto afectaba a la hora definir qué valor debía tener el umbral de clasificación, por lo que se decidió implementar un algoritmo de calibración para que de forma más automática, detectase el valor óptimo para cada sujeto. Este proceso de calibración consistía en pedirle al sujeto que, durante unos segundos, alternara flexión y extensión de los músculos del antebrazo anterior derecho. El algoritmo detectaba los valores máximo y mínimo y con ello determinaba donde debía posicionar el umbral.

Algoritmo de calibracion del umbral

% Definiciones iniciales

- Inicializamos dos variables (valor_max y valor_min) en donde almacenaremos los valores máximo y mínimo que se obtengan en un periodo de tiempo prefijado.
- Asignar un pin analógico a la placa de adquisición de señal

% En el bucle principal

Durante (30 segundos) hacer

- Leer dato del pin analógico correspondiente

Si (valor > valor_max) entonces

valor_max = valor

sino si (valor < valor_min) entonces

valor_min = valor

Fin Si

Fin Hacer

% El coeficiente 0.3 es un parámetro que se obtuvo experimentalmente. Dado que el incremento de la diferencia de potencial en función la intensidad del movimiento no es lineal, este parámetro sirve como ajuste o regulador.

Umbral = 0.3 × (valor_max - valor_min);

6.5. Tabla de tiempos

Las tareas y tiempos referentes al desarrollo de esta fase del proyecto fueron:

<i>Tarea</i>	<i>Horas</i>
Estado del arte	40
Programación de la rutina de interrupción de reloj en Arduino para el muestreo a 500Hz	10
Soldadura de pines y cables para integrar la placa en Arduino	5
Programación del control en Matlab	20
Pruebas	2
Total	77

Cuadro 2: Tabla de tiempos de la fase de control con EMG

6.6. Alcance

Los objetivos planteados en esta fase del proyecto se superaron satisfactoriamente. Se realizó una batería de pruebas de caja negra para validar si el sistema hacía lo que debía hacer. Se contó con la ayuda de 4 sujetos a los que se les pidió que, durante dos sesiones de 2 minutos cada una, alternaran mano abierta y mano cerrada cada 20 segundos. El funcionamiento fue el correcto y se obtuvo una efectividad del 100 %.

En este vídeo de youtube <https://youtu.be/DZixSyP47aM> se puede ver una demo del control implementado.

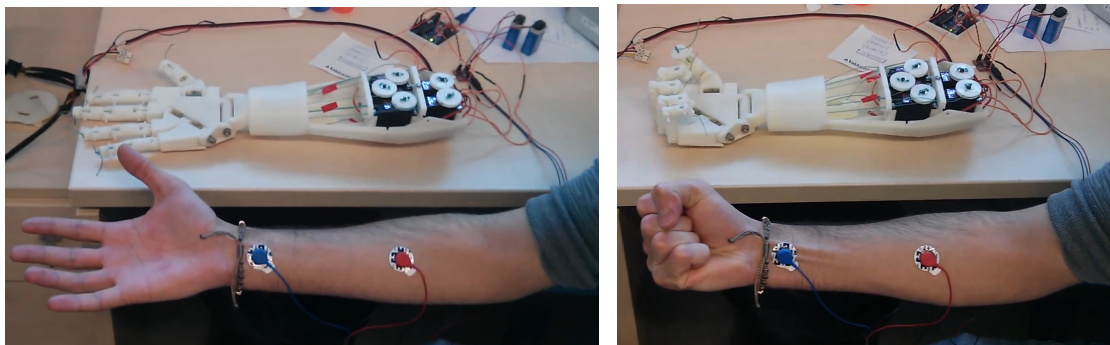


Figura 6.15: Pruebas de control con EMG del brazo robótico

7. Control con EEG

Dado que éste es un tema actual de investigación, y que en la comunidad científica aún no existe consenso respecto a qué técnicas son las mejores, así como los límites que ofrece esta tecnología, en esta fase se procedió a la evaluación y puesta a punto de dos clasificadores de imaginación de movimiento, a través de señales electroencefalográficas (EEG), con el objetivo principal de dejar listo el sistema para que se pudiesen enviar órdenes al brazo robótico construido anteriormente.

Objetivos

Los objetivos marcados para esta fase fueron: (1) conseguir una copia local de la base de datos para poder trabajar offline; (2) implementar una etapa de filtrado de artefactos en la señal EEG para mejorar la calidad señal/ruido; (3) implementar una etapa de extracción de características de la señal filtrada relacionadas con la tarea en concreto que se quiso analizar (apertura y cierre de mano derecha); (4) implementar una etapa de entrenamiento de dos clasificadores (SOM y LDA) para una posterior evaluación y comparación de resultados; (5) Las implementaciones debían ser compatibles con la toolbox de Matlab Fieldtrip, que ofrece una serie de métodos y funciones para analizar y visualizar datos de señales EEG.

En la figura 7.16 se puede ver la pipeline del sistema, ordenada en tres grandes bloques secuenciales que se corresponden con las diferentes etapas de procesado de la actividad cerebral hasta obtener los resultados de clasificación. Los módulos correspondientes a cada etapa fueron diseñados de forma que facilitase la escalabilidad, dado que como todavía es un campo que se está investigando, continuamente van surgiendo diferentes métodos de análisis que en un futuro podría ser interesante incorporar en el sistema.

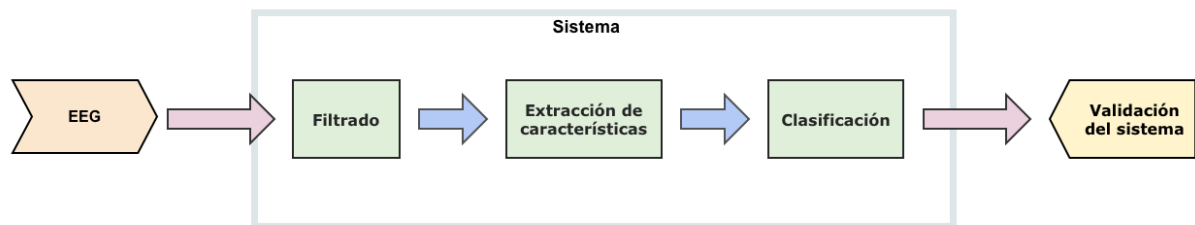


Figura 7.16: Esquema general del sistema de clasificación y sus diferentes etapas.

Retos/Dificultades encontradas

Esta fase del proyecto sin duda fue la más costosa debido a la falta de formación en este campo tan amplio. Se invirtió mucho tiempo en explorar el estado del arte para entender tanto qué metodologías se empleaban para resolver este tipo de problemas, como las técnicas que se utilizaban. Estas técnicas no están incluidas en el plan de estudios de Ingeniería Informática, por lo que hubo que cursar complementos de formación de otras carreras para solventar este déficit.

Así por ejemplo, hubo que cursar asignaturas de Ingeniería de Telecomunicaciones relacionadas con procesamiento digital de señales biológicas o Neurocomputación, y como la base de estas asignaturas tampoco se corresponde con la que se ofrece en Ingeniería Informática, hubo que complementar esa formación para poder asimilar bien los conceptos. Hubo que aprender a utilizar la Toolbox de Matlab Fieldtrip., que si bien es una herramienta muy útil, no es trivial su uso.

7.1. Córtex cerebral y EEG.

¹ El córtex cerebral forma parte del sistema nervioso central (SNC), y las diferentes regiones que lo integran son las responsables del procesamiento de funciones vitales como son la percepción, el movimiento voluntario, el habla o el aprendizaje. Tiene un grosor de unos 2-3 mm. Pese a ser una delgada capa, su superficie está fuertemente circunvolucionada, y su área total se estima en unos $2,5m^2$. El córtex está constituido por dos hemisferios - izquierdo y derecho - que están separados por una cisura sagital en la línea media. Cada hemisferio está subdividido en cuatro lóbulos: frontal, parietal, occipital y temporal.

Los movimientos voluntarios son principalmente controlados por el área del lóbulo frontal justo anterior a la hendidura central - el córtex motor. Las tareas que requieren un control notable de músculos, e.g., habla, ciertos gestos faciales, o el movimiento de los dedos, están asociadas con la subárea más grande del córtex motor. La información sensorial es procesada en varias partes de los lóbulos: el córtex auditivo está localizado en la parte superior del lóbulo temporal, el córtex visual está localizado en la parte posterior del lóbulo occipital y el córtex somatosensorial está localizado en la parte posterior de la hendidura central del lóbulo parietal, ver Figura 7.17.

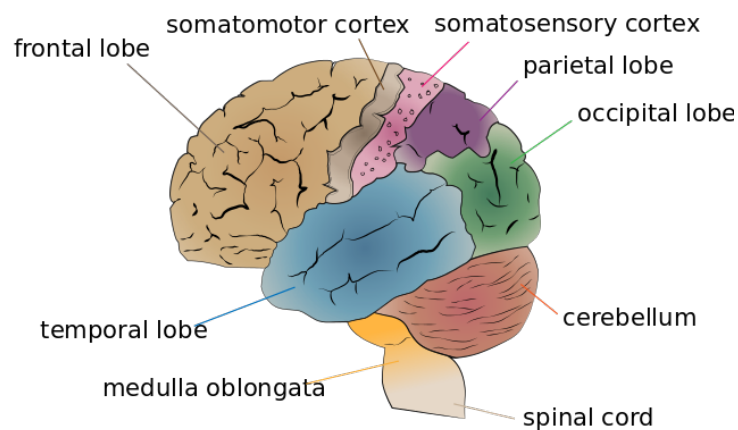


Figura 7.17: El córtex cerebral

La actividad eléctrica colectiva del córtex cerebral es referida usualmente como ritmo, porque

¹Esta sección está inspirada en el libro de Leif Sörnmo y Pablo Laguna, “Bioelectrical signal processing in cardiac and neuro-logical applications”, para más información consultar los capítulos 2 (“The Electroencephalogram - A Brief Background”) [14] y 3 (“EEG Signal Processing”) [15]

las señales obtenidas en la medición suelen exhibir un comportamiento oscilatorio, repetitivo. La actividad de una única neurona no puede ser medida en la superficie del cuero cabelludo debido a las gruesas capas de tejidos (fluidos, huesos y piel) que atenúan la señal eléctrica cuando se propaga hacia el electrodo. Sin embargo, la actividad conjunta de millones de neuronas del córtex, a una profundidad de varios milímetros, produce un campo eléctrico lo suficientemente fuerte para ser medido en el cuero cabelludo. Al registro de dicha actividad bioeléctrica se le conoce como electroencefalograma (EEG). El campo eléctrico es generado principalmente por corrientes que fluyen durante la excitación sináptica de las dendritas, los potenciales excitatorios postsinápticos.

La diversidad de ritmos en el EEG es enorme y depende, entre otras muchas cosas, del estado mental del sujeto, como puede ser el grado de atención o el sueño. Los ritmos suelen caracterizarse por su rango de frecuencias y su amplitud relativa. Dicha amplitud está relacionada con el grado de sincronización en el cual las neuronas del córtex interactúan. Una excitación sincrónica de un grupo de neuronas produce una señal de gran amplitud en el cuero cabelludo debido a que las señales originadas por neuronas individuales se sumarán coherentemente en el tiempo. Repeticiones de excitaciones sincrónicas resultan en una señal EEG rítmica. Por otro lado, una excitación asincrónica de un grupo de neuronas resulta en un EEG con apariencia irregular con formas de onda de baja amplitud. En ambos casos (sincronía y asincronía), la excitación puede perfectamente implicar un número idéntico de neuronas, pero, dependiendo de la dispersión temporal del input neuronal, diferentes amplitudes en el EEG aparecen.

Las señales registradas en el cuero cabelludo suelen alcanzar amplitudes en el rango de unos pocos microvoltios hasta aproximadamente unos $100 \mu\text{V}$, y se suelen mover en el rango de frecuencias entre 0.5-40 Hz. Los ritmos del EEG son convencionalmente clasificados en 5 bandas de frecuencia diferentes. La interpretación de estas bandas en términos de normal o anormal es relativa y depende de la edad y el estado mental del sujeto. Por ejemplo, el EEG de un recién nacido es completamente distinto del de un adulto. Las bandas de frecuencia que se muestran a continuación son demasiado rígidas, pero aun así permiten una útil categorización clínica de los diferentes ritmos:

- **Ritmo Delta, < 4 Hz**
- **Ritmo Theta, 4-7 Hz**
- **Ritmo Alpha, 8-13 Hz**
- **Ritmo Beta, 14-30 Hz**
- **Ritmo Gamma, > 30 Hz**

Uno de los aspectos cruciales en procesamiento de señales biomédicas es adquirir conocimiento acerca del ruido y los artefactos que están presentes en la señal, para minimizar su influencia. Por lo tanto, el procesamiento de artefactos constituye uno de los puntos clave en los análisis de señales biomédicas, y es igual de importante si las señales son originarias del cerebro, del corazón o de otra fuente eléctrica del cuerpo humano. En los registros de EEG, una amplia variedad de tipos de artefactos pueden ocurrir, algunos de los cuales son fáciles de identificar

con algoritmos simples, mientras que otros poseen características similares al propio EEG, lo que los hace difíciles de distinguir.

Una clasificación muy útil de los artefactos es aquella basada en su origen, i.e., origen fisiológico o tecnológico. Esta categorización es además aplicable a otras señales bioeléctricas. Algunos de los artefactos más comunes son:

Movimientos oculares y parpadeos. El movimiento ocular genera actividad eléctrica - el electrooculograma (EOG) - que es lo suficientemente fuerte como para ser claramente visible en el EEG. El EOG refleja la diferencia de potencial entre la córnea y la retina, la cual cambia durante el movimiento del ojo. Otro artefacto común es el causado por el movimiento de los párpados ("parpadeos") que también influye en la diferencia de potencial entre la córnea y la retina. Los artefactos por parpadeos normalmente producen cambios más abruptos en la forma de onda que los movimientos oculares, y en consecuencia, contienen más componentes de alta frecuencia.

Actividad muscular. Otro tipo de artefacto común es aquel generado por la actividad eléctrica de los músculos. Este tipo de artefacto ocurre, e.g., al tragar, gesticular, fruncir el ceño, masticar, hablar o durante un ataque de hipo. La forma total de la señal EMG depende del grado de contracción muscular: una contracción leve produce una secuencia de picos de baja amplitud, mientras que un incremento en la fuerza de contracción reduce la distancia entre picos por lo que el EMG se asemeja más a una señal con características de ruido "coloreado". En términos de procesamiento de artefactos, las propiedades del espectro de frecuencias del EMG son mucho menos favorables que las asociadas a movimientos oculares, porque hay un solapamiento considerable en la banda Beta, en el rango 15-30 Hz.

Actividad cardíaca. La actividad eléctrica del corazón, reflejada en el electrocardiograma (ECG), puede interferir con el EEG. Aunque la amplitud de la actividad cardíaca es normalmente baja en el cuero cabelludo en comparación con el EEG (1-2 y 20-100 μV , respectivamente), puede dificultar la lectura del EEG considerablemente en ciertas localizaciones de los electrodos y también según la corpulencia de los sujetos, e.g., pequeño, vigoroso, cuello grueso.

Electrodos y equipamiento. Este tipo de artefactos técnicos no es único del EEG, sino que puede ocurrir en cualquier adquisición de señales bioeléctricas en la superficie corporal. Si un electrodo se despega y deja de hacer contacto con la superficie de la piel, produce cambios en el potencial, que se suele manifestar como un cambio abrupto en la línea base, seguido de una lenta y gradual regresión a la línea base original. El cable del electrodo, que conecta éste con el equipo de adquisición, también puede ser una fuente de artefactos. Si no está lo suficientemente protegido puede ser afectado por campos electromagnéticos causados por corrientes originadas en equipos cercanos o en tomas de red eléctrica cercanas. Como resultado, interferencias a 50/60 Hz de la red eléctrica pueden contaminar el EEG.

El enfoque común en el procesamiento de artefactos es, primero, estimar el ruido $\nu(n)$, ya sea desde la propia señal adquirida, como desde otra señal de referencia, y después sustraer dicho ruido de la señal original. Este enfoque asume implícitamente que la señal original se puede descomponer en una suma entre la actividad cerebral $s(n)$ y el ruido $\nu(n)$,

$$x(n) = s(n) + \nu(n) \quad (1)$$

Al mismo tiempo que esta asunción ha sido aceptada, obviamente tiene sentido estimar $s(n)$ sustrayendo una aproximación del ruido $\hat{\nu}(n)$. Este tipo de modelo está asociado no solo con métodos de sustracción, sino además subyace una reducción del ruido a través de un filtrado lineal de la señal original.

La gran popularidad del modelo aditivo en (1) se explica por su simplicidad y la gran cantidad de métodos desarrollados que hay disponibles para realizar una estimación óptima de $s(n)$. Sin embargo, hay que saber que este tipo de modelo no ha de ser necesariamente el más apropiado, sino que existen otros modelos como el multiplicativo, que asume que señal y ruido interactúan en el registro como factores de un producto:

$$x(n) = s(n)\nu(n) \quad (2)$$

7.2. Base de datos

Para llevar a cabo este trabajo, dado que no se disponía de un equipo de adquisición para poder realizar mediciones propias, se utilizó la base de datos de EEG disponible en Physionet [16] *EEG Motor Movement/Imagery Dataset*, base de datos que fue creada y aportada por los desarrolladores de BCI2000 [17], sistema que usaron para realizar estos registros. Esta base de datos consiste en más de 1500 registros de EEG, obtenidos de 109 voluntarios. Los sujetos realizaron diferentes tareas motoras mientras se les registraba el EEG a través de 64 electrodos colocados en el cuero cabelludo.

Los registros son proporcionados en ficheros con formato EDF (European Data Format) conteniendo cada uno 64 señales de EEG (una por electrodo), y un canal adicional con anotaciones relacionadas con eventos del experimento. El EEG fue registrado a una frecuencia de muestreo de 160 Hz y usando el sistema internacional 10-10 [18] (excluyendo los electrodos Nz, F9, F10, FT9, FT10, A1, A2, TP9, TP10, P9, y P10), como se muestra en la Figura 7.18.

Por cada sujeto se realizaron catorce registros de EEG: primero se realizó un registro de un minuto donde el sujeto permanecía en reposo con los ojos abiertos; otro registro de un minuto en donde el sujeto se mantenía en reposo pero esta vez con los ojos cerrados (tanto el registro de ojos abiertos como el de ojos cerrados, son clave para según que tipo de análisis posterior se quiera hacer), y después, se realizaron doce registros de dos minutos cada uno en los cuales al sujeto se le pidió que llevase a cabo tres repeticiones de cada una de las cuatro tareas que exponemos a continuación (cada registro de dos minutos corresponde con una repetición de una tarea en concreto):

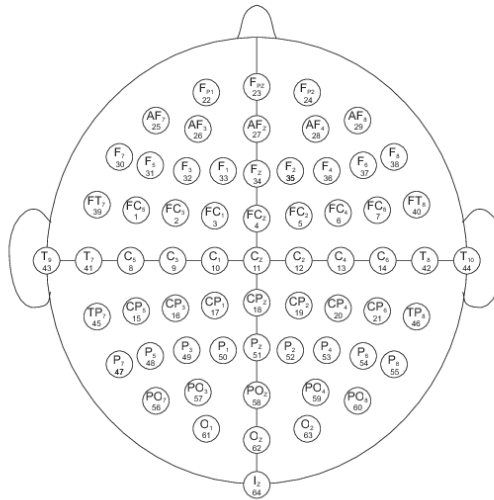


Figura 7.18: Montaje usado en la adquisición.

1. Un objetivo aparecía o bien a la izquierda o bien a la derecha de la pantalla. El sujeto abría o cerraba el puño del correspondiente lado hasta que el objetivo desaparecía. Después el sujeto se relajaba.

2. Un objetivo aparecía o bien a la izquierda o bien a la derecha de la pantalla. El sujeto **imaginaba** que abría o cerraba el puño del correspondiente lado hasta que el objetivo desaparecía. Después el sujeto se relajaba.

3. Un objetivo aparecía o bien arriba o bien abajo de la pantalla. El sujeto abría o cerraba, bien ambos puños (si el objetivo aparecía arriba) o bien ambos pies (si el objetivo aparecía abajo) hasta que desapareciese. Después el sujeto se relajaba.

4. Un objetivo aparecía o bien arriba o bien abajo de la pantalla. El sujeto **imaginaba** que abría o cerraba, bien ambos puños (si el objetivo aparecía arriba) o bien ambos pies (si el objetivo aparecía abajo) hasta que desapareciese. Después el sujeto se relajaba.

No se especifica si, para cada sujeto, los catorce registros se realizaron en la misma sesión o de forma discontinua, por lo que no podemos asumir, a priori, hechos como que el sujeto no se quitó y volvió a poner los electrodos en algún momento, que no transcurrieron horas o días

entre registros, o incluso si el orden de registro fue el que indica la nomenclatura de los ficheros.

7.3. Preprocesado

Durante los análisis, se detectaron errores en el registro y anotaciones de varios sujetos (sujetos nº 43, 88, 89, 92, 100 y 104) por lo que no fueron tomados en cuenta. Al resto de registros, se aplicó un filtro paso-banda Butterworth de orden 4 en $[1,70]$ Hz y un filtro elimina-banda a 60Hz para eliminar las interferencias de línea de alimentación.

Después, se implementaron una serie de algoritmos en Matlab con el objetivo de abordar el filtrado de artefactos de la señal EEG. Dado que existen múltiples técnicas para este cometido y todavía no hay consenso en la comunidad científica respecto a la conveniencia o no del uso de algunas de ellas, se optó por la implementación de tres métodos correspondientes al filtrado de tres tipos de artefactos distintos pero se dejó definida una estructura modular para facilitar la integración de nuevos métodos. En la fig. 7.19 se puede ver un diagrama con los tipos de artefactos tenidos en cuenta así como los métodos implementados. A continuación se expondrán cada uno de ellos.

Lo primero que se realizó fue un análisis de componentes independientes (ICA [19], [20], [21]), para separar aquellas componentes de la señal correspondientes con la actividad neuronal, de aquellas que contenían artefactos de algún tipo. Después se aplicaron diferentes criterios para discriminar las componentes, en función del tipo de artefacto que se analizaba.

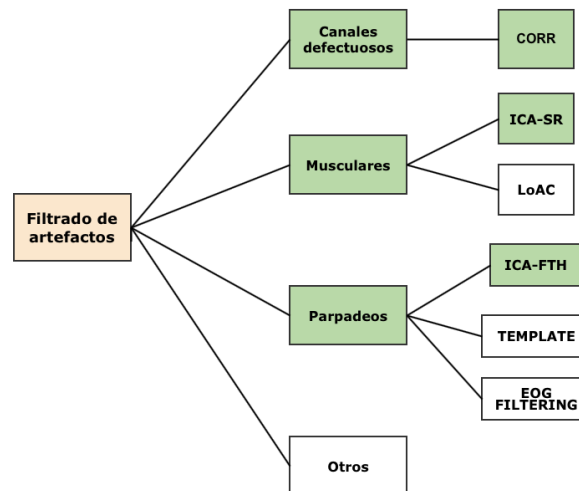


Figura 7.19: Módulo de filtrado de artefactos. En color verde se muestran aquellos componentes que se han implementado en este proyecto, y se muestran sin colorear ejemplos de otros posibles componentes que se podrían implementar en un futuro.

7.3.1. Análisis de Componentes Independientes

Inicialmente se realizó un análisis de componentes independientes con el algoritmo FastICA [22]. Mediante este análisis se asume que las n señales registradas (una por canal), x_1, x_2, \dots, x_n , son una combinación lineal de n fuentes independientes:

$$x_j(t) = a_{j1}s_1(t) + a_{j2}s_2(t) + \dots + a_{jn}s_n(t) \quad (4)$$

o, usando notación matricial, $x = As$. Así pues, el objetivo es encontrar la matriz de desmezclado W tal que $s = Wx$. Una vez se consigue esa matriz de desmezclado, es posible analizar algunas características temporales, frecuenciales y topográficas de cada una de las fuentes por separado. Para más detalle ver Anexo II [10].

7.3.2. Características de los canales defectuosos

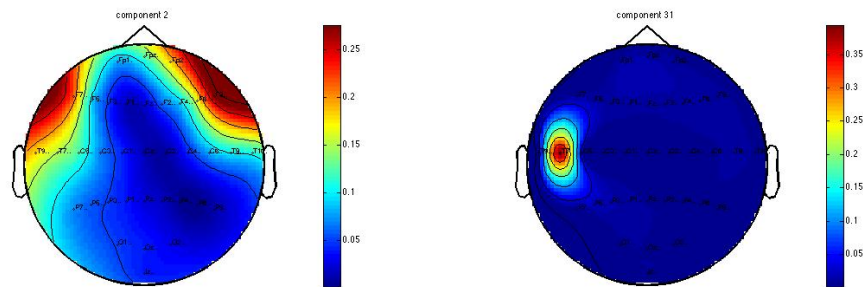
Durante el registro de la actividad cerebral, es posible que uno o varios electrodos se despeguen del cuero cabelludo. Esto puede producir fluctuaciones de gran amplitud (en relación con las producidas por la actividad neuronal) y una no correlación con respecto al resto de canales. Según Maximilien Chaumon et al. [23], ICA aísla en una única componente este tipo de artefactos. Una componente que captura un canal defectuoso muestra una topografía muy focalizada en dicho canal, y una fuerte correlación entre ambos.

Un primer algoritmo para detectar este tipo de componentes consistió en calcular el coeficiente de correlación de Pearson entre cada par Canal-Componente, y colocar un umbral “alto” mediante el cual se pudiese detectar una fuerte relación entre componentes y canales. Sin embargo, en las pruebas resultó que este criterio, si bien era necesario, no era suficiente debido a que algunas componentes mostraban altas correlaciones con varios canales (ver fig.7.20a). Por ello, se decidió añadir un segundo criterio: para cada componente que sobrepasase el umbral de correlación, dicha correlación debía ser suficientemente grande con respecto al resto de correlaciones entre la misma componente y los demás canales. Esto se hizo calculando a cuantas desviaciones típicas se encontraba dicha correlación, con respecto a la media. Así pues, otro umbral hubo que fijar para determinar cuando un valor era lo suficientemente significativo como para determinar que se correspondía con un canal defectuoso. Y para generalizar el algoritmo, se decidió normalizar todas las variables según la ecuación:

$$z = \frac{x - \mu_X}{\sigma_X} \quad (5)$$

donde μ_X y σ_X son respectivamente la media y desviación típica del conjunto de correlaciones Canal-Componente.

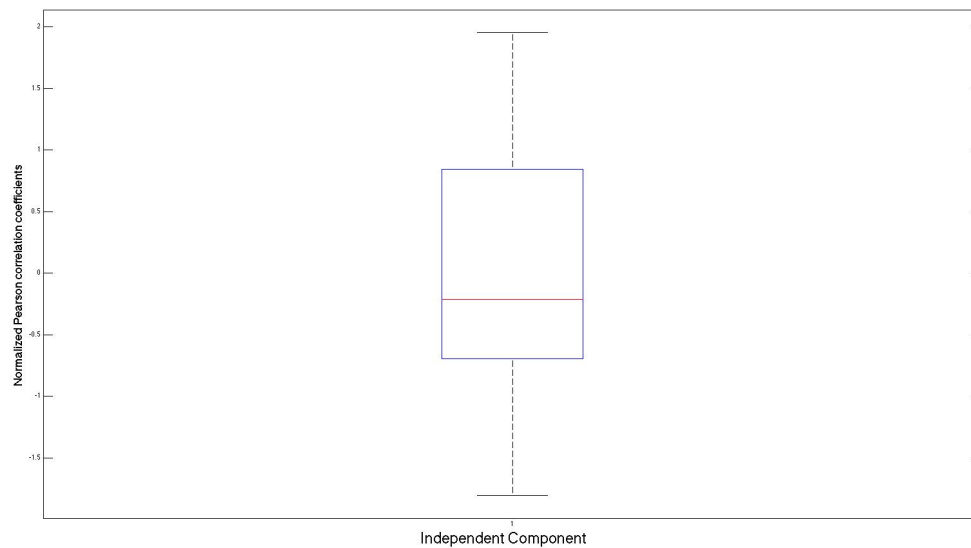
En la fig. 7.20b se puede ver un ejemplo de una componente independiente detectada como canal defectuoso una vez introducido este segundo criterio en el algoritmo, que se corresponde con el electrodo T7 según el montaje de la fig. 7.18.



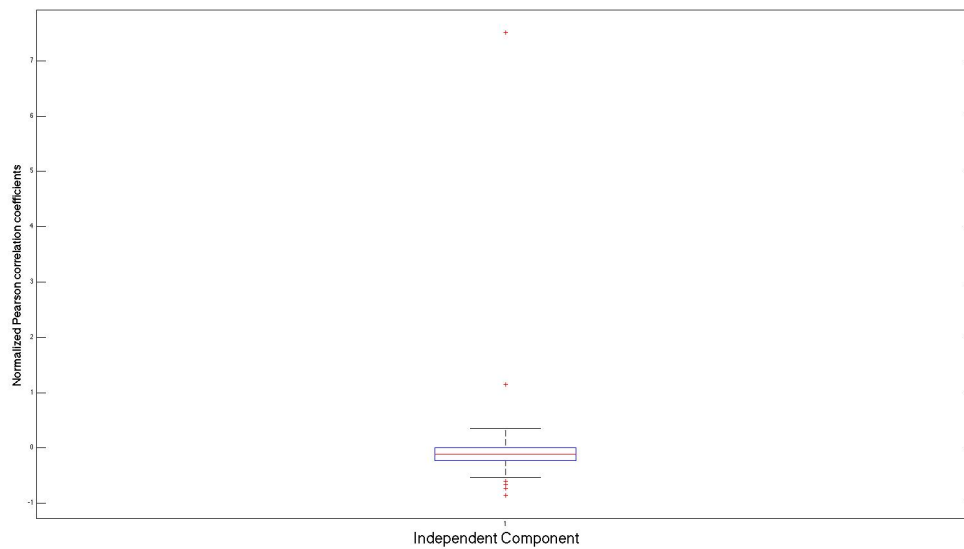
(a) Componente independiente detectada erróneamente como canal defectuoso (b) Componente independiente detectada correctamente como canal defectuoso

Figura 7.20: Topoplots durante la clasificación de canales defectuosos

En la fig. 7.21 se pueden ver dos gráficas de dispersión correspondientes a las mismas componentes que en la fig. 7.20, es decir, la fig. 7.21a se corresponde con una componente que fue detectada como canal defectuoso antes de introducir el criterio de dispersión, y la fig. 7.21b se corresponde con una componente que fue detectada como canal defectuoso después de introducir el criterio de dispersión. Los valores más altos se corresponde con altos índices de correlación entre Canal y Componente. Se puede apreciar como en la segunda gráfica el valor máximo está a más de 7 desviaciones típicas de la media, mientras que en la primera gráfica la dispersión es mucho menor.



(a) Boxplot del conjunto de coeficientes de correlación entre una componente independiente (detectada erróneamente como canal defectuoso) y todos los canales.



(b) Boxplot del conjunto de coeficientes de correlación entre una componente independiente (detectada correctamente como canal defectuoso) y todos los canales. El valor atípico que sobrepasa las siete desviaciones típicas se corresponde con el coeficiente de correlación entre la componente y el canal defectuoso.

Figura 7.21: Boxplots referentes a los coeficientes de correlación entre CIs y canales detectados como defectuosos.

7.3.3. Características de los artefactos musculares

Según Junshui Ma et al. [24], después de estudiar decenas de miles de componentes de señales EEG, una característica común que observaron de las componentes musculares fue que algunos segmentos correspondientes a altas frecuencias de la densidad espectral de potencia (PSD por sus siglas en inglés) se mantenían asintóticos o incluso aumentaban, mientras que las PSDs de otras componentes de señal EEG casi siempre decrecían en el rango de frecuencias por encima de 40 Hz. Una simple explicación biofísica de esta observación es que, las señales generadas en la corteza cerebral, antes de ser capturadas por los electrodos en la superficie del cuero cabelludo tienen que pasar varias capas de fluidos, tejido y hueso, que actúan como un filtro paso-bajo que atenúa la potencia de la señal, afectando proporcionalmente a sus componentes frecuenciales (altas frecuencias sufren mayor atenuación mientras que bajas frecuencias sufren menor atenuación). En contraste, la actividad muscular ocurre en la superficie de la cabeza, y por lo tanto se mantienen considerablemente las componentes de alta frecuencia. Estas observaciones van en la línea de Goncharova et al. [25].

El método que proponen Junshui Ma et al. [24] consiste, para cada componente independiente, primero en calcular el que denominan Ratio Espectral en ICA (ICA-SR), definido en la ecuación (6) y posteriormente aplicar un umbral que indicará si la componente corresponde con un artefacto muscular o no. Este ratio es una relación entre las altas y las bajar frecuencias, que si toma valores altos indicará que las componentes de altas frecuencias mantienen un nivel superior al normal y por lo tanto es un síntoma de que la componente puede contener un artefacto muscular.

$$\ln\left[\frac{H(S_i(f))}{L(S_i(f))}\right] \begin{cases} \leq \Delta \rightarrow \text{NO es un artefacto muscular} \\ > \Delta \rightarrow \text{SI es un artefacto muscular} \end{cases} \quad (6)$$

donde $S_i(t)$ es el espectro de potencia de la fuente $s_i(t)$;

$$H(S_i(f)) = \frac{1}{f \in [40, 60]} \sum_{f \in [40, 60]} S_i(f) \quad (7)$$

denota la media del espectro de potencia en la banda de altas frecuencias entre 40 y 60 Hz;

$$L(S_i(f)) = \underset{f \in [4, 30]}{\text{mín}} [\mathbf{F}_{smooth} \otimes S_i(f)] \quad (8)$$

se corresponde con el valor más pequeño del espectro de potencia en la banda de bajas frecuencias entre 4 y 30 Hz, \mathbf{F}_{smooth} se corresponde con un filtro de suavizado de media móvil y \otimes es el operador de convolución.

El umbral Δ es un parámetro del algoritmo de clasificación que debe definirse al comienzo. Si $\Delta > 0$ quiere decir que $H(S_i(f)) > L(S_i(f))$. Este parámetro, siguiendo las recomendaciones del artículo, y realizando pruebas experimentales, se fijó a 0.2. Si una componente mostraba un ratio superior a este valor, se etiquetaba como artefacto muscular para después ser eliminada.

7.3.4. Características de los parpadeos

Según Maximilien Chaumon et al. [23], las componentes independientes que capturan los parpadeos oculares presentan una distribución topográfica característica, centrando la actividad en los electrodos del lóbulo frontal más próximos a los ojos. Estos artefactos producen variaciones abruptas de gran amplitud, mucho mayores que las ocasionadas por la actividad neuronal, y en instantes concretos del curso de la señal. Una persona con experiencia puede detectar en que instantes se han producido parpadeos con una simple inspección visual de la señal registrada.

Aprovechando la focalización que presentan este tipo de artefactos en su distribución topográfica, se decidió implementar el siguiente método:

1. Aplicar FastICA para obtener las matrices de mezclado y desmezclado de componentes
2. Analizar cada componente por separado, anulando los coeficientes del resto de componentes en la matriz de mezclado
3. Fijar un umbral (en nuestro caso se decidió poner a $100\mu\text{V}$)
4. Analizar si en los canales frontales Fp1 y Fp2 se sobrepasa dicho umbral
5. Si se sobrepasa, considerar que esa componente contiene artefactos y anular sus pesos en la matriz de mezclado.
6. Una vez analizadas todas las componentes independientes, con la nueva matriz de mezclado reconstruir el EEG.

Para ilustrar el efecto de este método de filtrado, en la figura 7.22 se puede ver la derivación Fp1-F3 (según el montaje 7.18) antes del filtrado de artefactos. Los cambios abruptos de gran amplitud se corresponden con parpadeos. En la figura 7.23 se puede ver la señal una vez se han eliminado los artefactos.

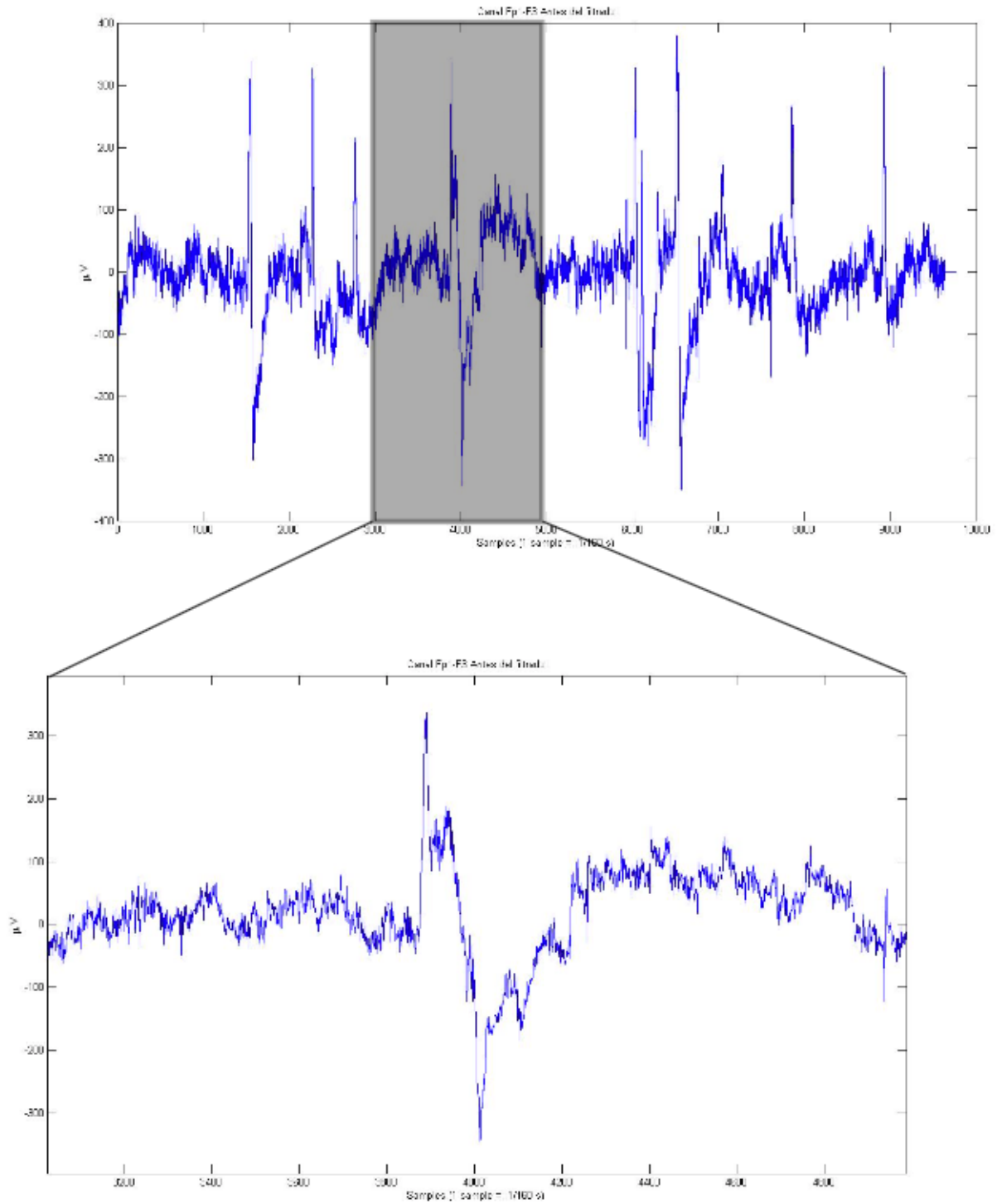


Figura 7.22: Canal Fp1-F3 en el dominio temporal, antes de realizar el filtrado de artefactos aplicando ICA. Eje de abscisas: muestras (1/160 s). Eje de ordenadas: $\mu Voltios$

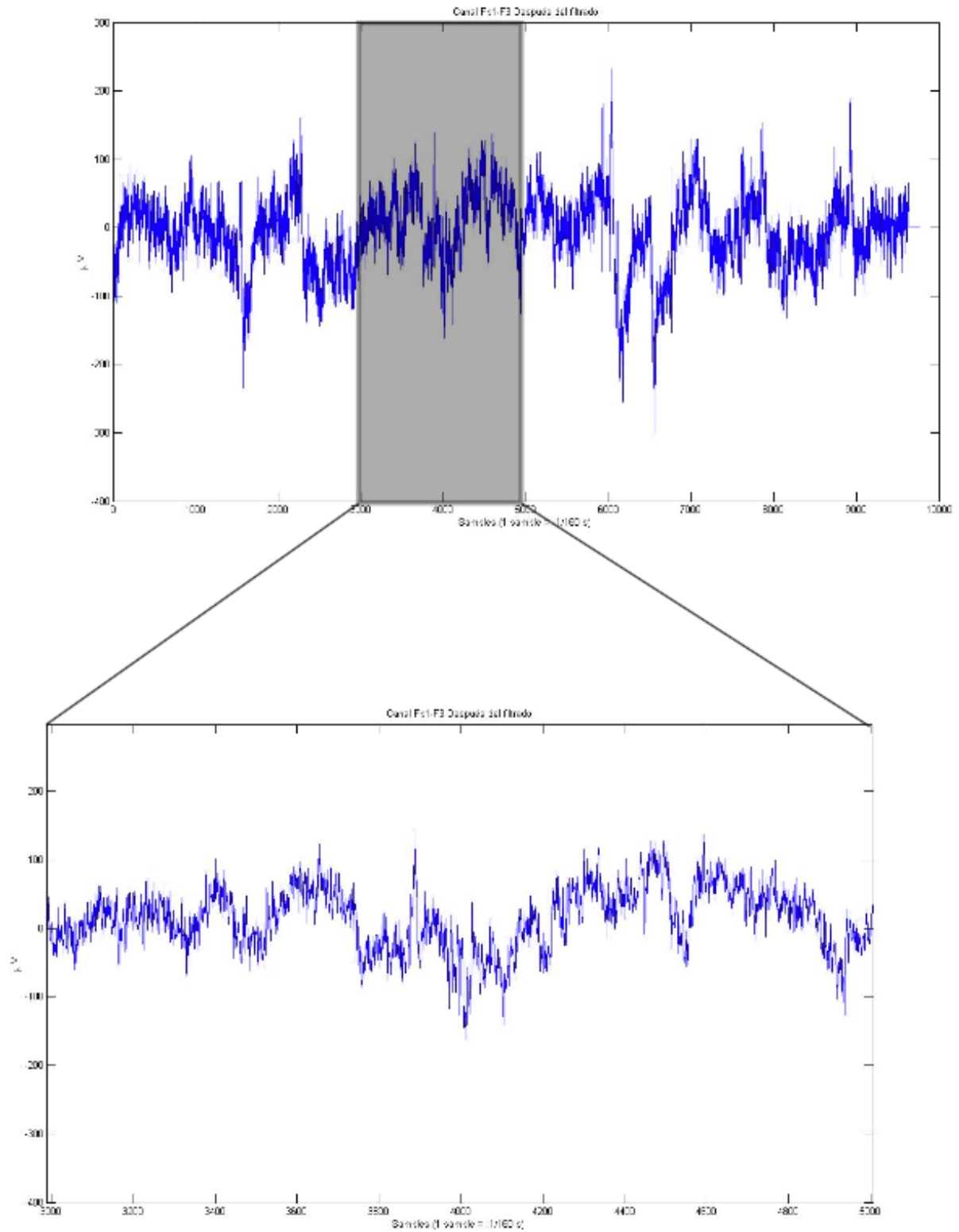


Figura 7.23: Canal Fp1-F3 en el dominio temporal, después de realizar el filtrado de artefactos aplicando ICA. Eje de abscisas: muestras (1/160 s). Eje de ordenadas: μV oltios

7.4. Extracción de patrones

Los registros de la base de datos (especificada en 7.2), salvo los correspondientes con las lecturas de ojos abiertos y ojos cerrados, mantienen la estructura especificada en la fig. 7.24. En un mismo registro se realizaron varias repeticiones de las tareas que correspondieran. Así, por cada registro, se tienen delimitados por eventos donde empieza y donde acaba cada tarea. A estas porciones de registro se les denomina trials. Los eventos indican qué tarea da comienzo, por lo que cada trial queda etiquetado con respecto al tipo de evento que le precede. Cada trial dura aproximadamente 4 segundos.

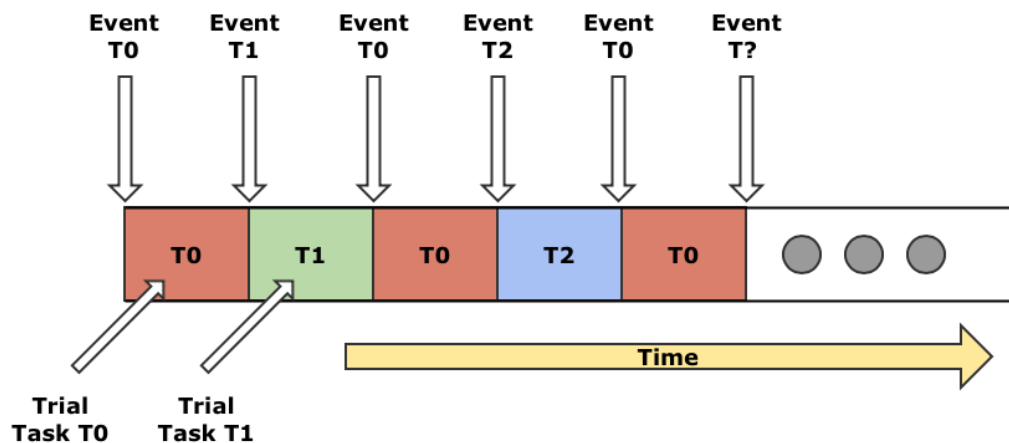


Figura 7.24: Estructura del fichero de registro de EEG.

Aunque la base de datos utilizada proporcionaba la posibilidad de analizar diferentes tareas, se optó por centrar este trabajo en intentar implementar un clasificador que detectase cuando el sujeto imaginaba que cerraba el puño derecho, y cuando el sujeto estaba en reposo. Estos estados se corresponden con la tarea 2 de la base de datos (series 4, 8 y 12). La correspondencia entre estados y etiquetas para la tarea 2 es la siguiente:

- Etiqueta T0: Reposo
- Etiqueta T1: Imaginar abrir/cerrar puño izquierdo
- Etiqueta T2: Imaginar abrir/cerrar puño derecho

por lo que en este trabajo se extrajeron características de los trials correspondientes a las etiquetas T0 y T2. Cada registro consta aproximadamente de 15 trials de tipo T0 y alrededor de 7 de tipo T2.

Dado que en la comunidad científica aún no están definidas cuáles son las mejores características para llevar a cabo la tarea que aquí se pretendía, y que este proyecto lo que busca es proporcionar herramientas para seguir con esa investigación, se decidió probar con unas características muy básicas y comprobar qué resultados ofrecían los clasificadores.

Los movimientos voluntarios resultan en una desincronización en las bandas alfa y beta, más concretamente en torno a la banda [10 20] Hz ([26] [27] [28] [29] [30] [31] [32]). Además, según Pfurtscheller et al. [33], la imaginación de movimiento de la mano activa la zona sensomotora primaria, con la relación invertida de que el hemisferio izquierdo se relaciona con la parte derecha del cuerpo, y el hemisferio derecho con la parte izquierda. Se decidió entonces seleccionar el canal C3-Ref para extraer características de la tarea llevada a cabo dado que ésta se centró en la imaginación de movimiento de la mano derecha.

Tomando la banda de [10 20] Hz como referencia, los pasos que se siguieron para extraer las características fueron los siguientes:

- 1 Se dividió la banda de frecuencia de [10 20] Hz en 4 sub-bandas, de anchura 5Hz, centradas en las frecuencias [10 12.5 17.5 20] Hz.
- 2 Se calculó la potencia media en cada una de las sub-bandas.

En resumen, por cada trial se extrajeron cuatro valores correspondientes a la potencia media de la señal en cada una de las bandas descritas anteriormente.

7.5. Métodos de clasificación

Los métodos de clasificación escogidos fueron los mapas auto-organizados de Kohonen y el análisis discriminante lineal.

7.5.1. SOM

En este apartado, se usó un tipo de red neuronal artificial llamado Mapa Auto-organizado [34]. El SOFM (Self Organizing Feature Map) propuesto por Teuvo Kohonen, tiene una arquitectura progresiva de dos capas (ver fig. 7.25). La segunda capa forma el mapa de rasgos, con una disposición rectangular o hexagonal de neuronas. Se comporta como una red K-WTA (K-Winners Take All, modelo competitivo de neuronas equivalente al K-clustering, donde K ganadoras se lo quedan todo).

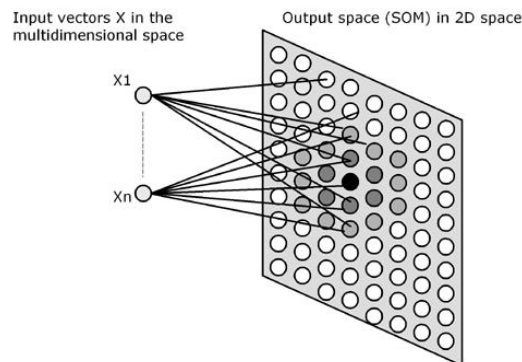


Figura 7.25: Capas representativas del SOM

Los SOFM proyectan de forma no lineal y topológicamente ordenado el espacio multidimensional de entrada sobre el mapa (ver fig. 7.26): entradas similares se corresponden con pesos w de neuronas próximas. El mapa representa con mayor fidelidad las dimensiones del espacio sensorial de mayor varianza (suelen coincidir con rasgos más importantes o las dos componentes principales según un análisis PCA). El entrenamiento consta de dos fases: la fase de despliegue del mapa, y la fase de ajuste fino.

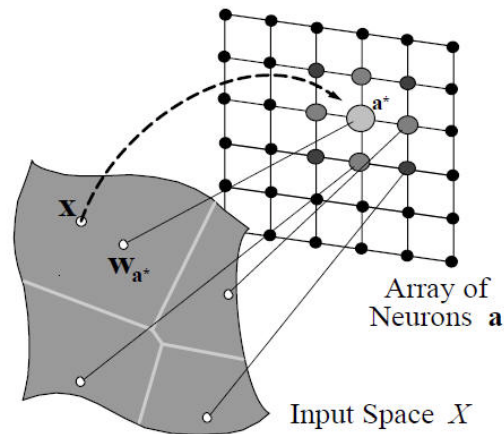


Figura 7.26: Proyección del espacio de entrada en el mapa de neuronas.

La implementación que se utilizó en este trabajo fue la correspondiente a la SOM toolbox para Matlab disponible en www.cis.hut.fi/projects/somtoolbox/.

7.5.2. LDA

El Análisis Discriminante Lineal (LDA) suele ser utilizado como técnica de reducción de dimensionalidad en una etapa previa de aplicaciones de reconocimiento de patrones, pero también es usado como clasificador. Es una generalización del discriminante lineal de Fisher. El objetivo es proyectar el conjunto de datos inicial en un espacio dimensional de menor orden aumentando la separabilidad entre las clases que se quieren distinguir.

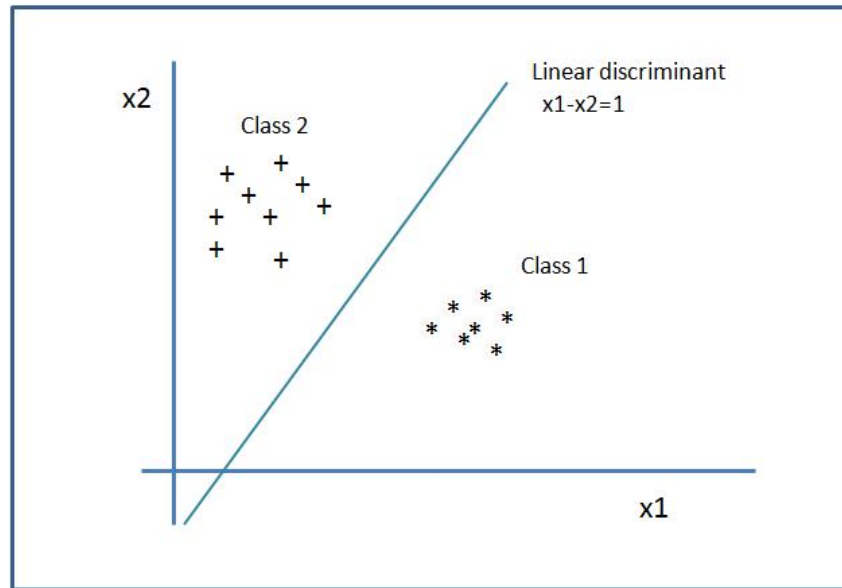


Figura 7.27: Ejemplo representativo de un discriminante lineal de dos clases

El enfoque de esta técnica es muy similar al Análisis de Componentes Principales (PCA), solo que, en vez de buscar los ejes que maximicen la variabilidad de los datos, este método busca los ejes que maximicen la separación entre clases. Así, lo que este método tratará de buscar, es un hiperplano que separe el conjunto de datos entre las clases de interés.

La implementación utilizada para esta parte fue la de Will Dwinnell y se puede encontrar en <http://www.mathworks.com/matlabcentral>.

7.6. Criterios de evaluación

Para la evaluación de los resultados, se escogieron los siguientes criterios: porcentaje de acierto (en cuanto a la identificación de a qué tarea correspondía cada trial), porcentaje de falsos positivos (de todos los trials en los que el sujeto estaba en reposo, cuantos se detectaban como que éste estaba imaginando que cerraba el puño derecho), porcentaje de no detección (de todos los trials en los que el sujeto imaginaba que cerraba el puño derecho, cuántos no se detectaban), y el índice Kappa como criterio principal para determinar cuando una clasificación había sido mejor o peor que otra. El índice Kappa [35] [36] es un estadístico que permite comparar el grado de concordancia entre dos observadores (en este caso clasificadores), cada uno de los cuáles etiqueta a N elementos en C clases mutuamente excluyentes. En este caso se compararon los clasificadores implementados con un clasificador ideal que identificaba correctamente cada trial con la tarea correspondiente. Cuanto más cercano a uno es el índice Kappa, mayor concordancia existe entre los clasificadores.

Además de estos índices se tuvo en cuenta a la hora de interpretar los resultados las matrices de confusión, ya que éstas permiten visualizar de una forma sencilla si el clasificador está confundiendo una clase con otra. Las columnas representan cada una de las clases existentes, y las filas las predicciones que el clasificador ha hecho de cada clase. En la diagonal de la matriz quedan representados el conjunto de valores que se han clasificado correctamente, mientras que fuera de la matriz se encuentran los elementos que el clasificador ha detectado que pertenecían a una clase, pero se correspondían con otra. En el cuadro 3 se puede ver un ejemplo de una matriz de confusión de un clasificador de tres clases sobre un conjunto de 140 elementos. En la fila 1 se puede ver que el clasificador ha detectado, de entre los 140 elementos, 35 como que pertenecían a la clase 1. De estos 35 elementos, 20 se han detectado correctamente, 10 pertenecían realmente a la clase 2, y 5 a la clase 3.

	Clase 1	Clase 2	Clase 3	Total Detectado
Detectada Clase 1	20	10	5	35
Detectada Clase 2	4	8	3	15
Detectada Clase 3	20	40	30	90
Total Real	44	58	38	140

Cuadro 3: Ejemplo de matriz de confusión.

7.7. Validación cruzada

A la hora de entrenar y validar los clasificadores, se realizó una partición de los datos para cada una de las fases, ya que no es conveniente realizar la validación con los mismos datos de entrenamiento. En este trabajo se decidió utilizar el 80 % de los trials para entrenamiento, y el restante 20 % para validación. En muchas ocasiones, este particionado condiciona el rendimiento del clasificador debido a la buena o mala calidad de los datos empleados para cada una de las fases. Por este motivo, se decidió aplicar el método de validación cruzada aleatoria (este método consiste en seleccionar de forma aleatoria, qué datos se utilizarán para entrenamiento, y qué datos se usarán para validación). Además, se realizaron 1000 repeticiones de este proceso para comprobar la dispersión de las medidas de rendimiento que ofrecían los clasificadores.

7.8. Resultados

A continuación se presenta un resumen de los resultados obtenidos. Dado que en el experimento, los sujetos debían repetir tres veces cada serie de tareas, intercalando cada repetición con la realización de otras tareas especificadas en el protocolo (ver descripción en [7.2]), es razonable pensar que el cansancio, el aprendizaje y soltura adquirida con las repeticiones y otro tipo de factores, pueden producir cambios en la actividad cerebral afectando a la variabilidad de los patrones extraídos, no solo inter-tarea, sino también intra-tarea, repercutiendo en el rendimiento de los clasificadores. Por ello, se decidió realizar dos análisis, uno teniendo en cuenta únicamente la primera serie de trials que el sujeto debía realizar respecto a esta tarea (R04 según el etiquetado de los autores), y otro teniendo en cuenta el conjunto de trials correspondientes a las 3 series (R04-R08-R12).

Como se ha dicho anteriormente, se realizó una validación cruzada aleatoria de 1000 repeticiones. En cada una de ellas se obtuvieron dos índices Kappa correspondientes al grado de clasificación de sendos modelos (LDA y SOM) en estudio. En lo que sigue se presenta un descriptivo de ambas distribuciones de índices Kappa obtenidas, casos mejor y peor, y su grado de dispersión.

En cada análisis, se realizó un test para comprobar si existían diferencias significativas entre las distribuciones de Kappas de cada modelo. Para cada distribución de Kappas se aplicó un test de Kolmogorov-Smirnov para determinar si éstas se ajustaban a una distribución normal. El resultado fue negativo ($\alpha = 0,05$) por lo que se descartó aplicar la prueba t-Student y en su lugar se realizó un test de suma de posiciones de Wilcoxon. En ambos análisis se encontraron diferencias significativas ($\alpha = 0,01$).

7.8.1. De una única serie

En el cuadro 4 se puede ver un descriptivo de la distribución de Kappas obtenida por ambos clasificadores en las 1000 repeticiones. En el cuadro 5 se tienen los resultados correspondientes con los mejores casos que ofrecieron ambos clasificadores, y en el cuadro 6 los correspondientes peores casos obtenidos en ambos modelos. Finalmente, en las figuras 7.28 y 7.29 están representadas mediante diagramas de caja, las distribuciones de los conjuntos de capas y los porcentajes de acierto respectivamente, de ambos modelos.

	Mejor	Peor	\bar{X}	σ
LDA	0,6041	0,4086	0,5057	$\pm 0,0276$
SOM	0,7327	0,3772	0,6079	$\pm 0,0527$

Cuadro 4: Descriptivo de los Kappas para una única serie

Caso Mejor

En este caso se pudo ver que SOM ofrecía mejores resultados que LDA en % de acierto y falsos positivos, pero en cambio este modelo estaba en torno a 6 puntos porcentuales por encima en cuanto a no detección de imaginación de movimiento.

	Reposo	Imaginación		Reposo	Imaginación
Detectado Reposo	74	28	Detectado Reposo	135	37
Detectado Imag.	81	126	Detectado Imag.	20	117

(a) Matriz de confusión del caso con mejor Kappa usando un clasificador LDA

(b) Matriz de confusión del caso con mejor Kappa usando un clasificador SOM

	Aciertos	Falsos positivos	No deteccion
LDA	64,72 %	52,26 %	18,18 %
SOM	81,55 %	14,81 %	24,03 %

(c) Comparación de resultados

Cuadro 5: Descriptivo del caso mejor en la clasificación de una única serie

Caso Peor

En este caso se pudo ver que SOM también ofrecía mejores resultados que LDA en % de acierto y falsos positivos, pero en cambio este modelo estaba en torno a 13 puntos porcentuales por encima en cuanto a no detección de imaginación de movimiento. Cabe destacar la diferencia notable de porcentajes de falsos positivos (63,23 % en LDA frente a 9,03 % en SOM).

	Reposo	Imaginación		Reposo	Imaginación
Detectado Reposo	57	59	Detectado Reposo	141	80
Detectado Imag.	98	95	Detectado Imag.	14	74

(a) Matriz de confusión del caso con peor Kappa usando un clasificador LDA

(b) Matriz de confusión del caso con peor Kappa usando un clasificador SOM

	Aciertos	Falsos positivos	No detección
LDA	49,19 %	63,23 %	38,31 %
SOM	69,58 %	9,03 %	51,95 %

(c) Comparación de resultados

Cuadro 6: Descriptivo del caso peor en la clasificación de una única serie

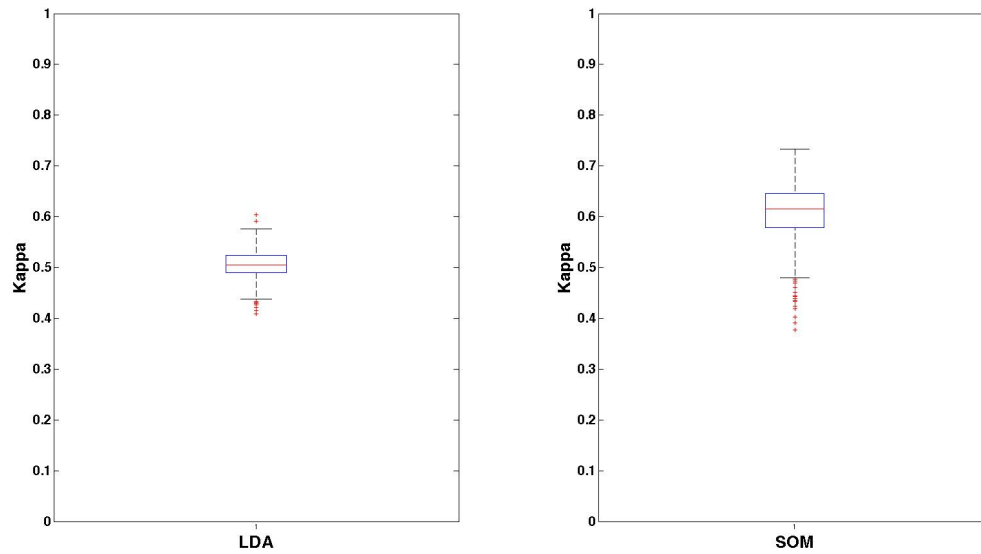


Figura 7.28: Diagramas de caja de las distribuciones de Kappas referentes a una única serie

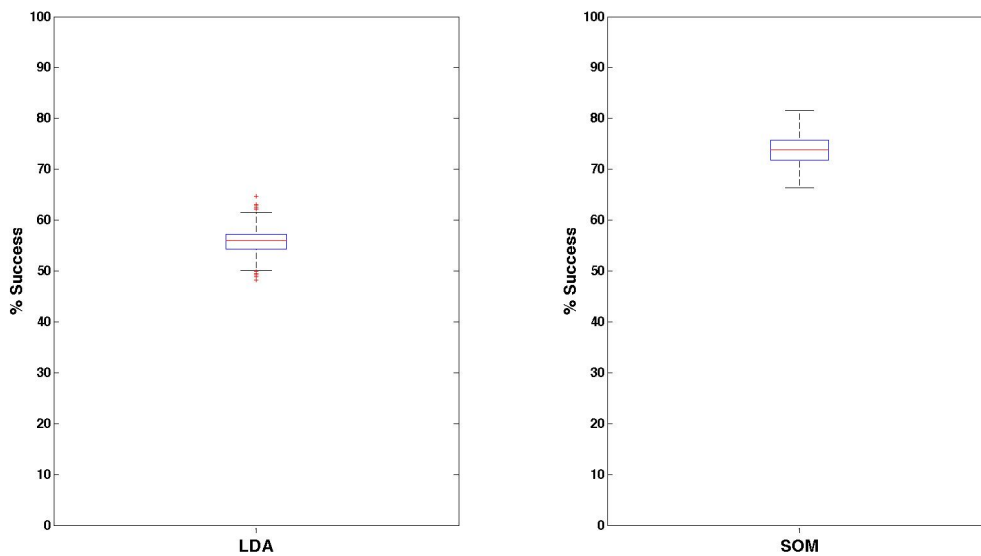


Figura 7.29: Diagramas de caja de las distribuciones de % de acierto referentes a una única serie

7.8.2. De todas las series

En el cuadro 7 se puede ver un descriptivo de la distribución de Kappas obtenida por ambos clasificadores en las 1000 repeticiones. En el cuadro 8 se tienen los resultados correspondientes con los mejores casos que ofrecieron ambos clasificadores, y en el cuadro 9 los correspondientes peores casos obtenidos en ambos modelos. Finalmente, en las figuras 7.30 y 7.31 están representadas mediante diagramas de caja, las distribuciones de los conjuntos de capas y los porcentajes de acierto respectivamente, de ambos modelos.

	Mejor	Peor	\bar{X}	σ
LDA	0,5534	0,4312	0,5015	$\pm 0,0169$
SOM	0,6468	0,3417	0,5476	$\pm 0,0401$

Cuadro 7: Descriptivo de los Kappas para todas las series

Caso mejor

En este caso se pudo ver que SOM ofrecía mejores resultados que LDA en % de acierto y falsos positivos, pero en cambio este modelo estaba en torno a 5 puntos porcentuales por encima en cuanto a no detección de imaginación de movimiento.

	Reposo	Imaginación		Reposo	Imaginación
Detectado Reposo	185	96	Detectado Reposo	335	119
Detectado Imag.	279	365	Detectado Imag.	129	342

(a) Matriz de confusión del caso con mejor Kappa usando un clasificador LDA

(b) Matriz de confusión del caso con mejor Kappa usando un clasificador SOM

	Aciertos	Falsos positivos	No detección
LDA	59,46 %	60,13 %	20,82 %
SOM	73,19 %	27,80 %	25,81 %

(c) Caso mejor Kappa

Cuadro 8: Descriptivo del caso mejor en la clasificación de todas las series

Caso peor

En este caso se pudo ver que SOM también ofrecía mejores resultados que LDA en % de acierto y falsos positivos, pero en cambio este modelo estaba en torno a 17 puntos porcentuales por encima en cuanto a no detección de imaginación de movimiento.

	Reposo	Imaginación
Detectado Reposo	166	162
Detectado Imag.	298	299

	Reposo	Imaginación
Detectado Reposo	351	244
Detectado Imag.	113	217

(a) Matriz de confusión del caso con peor Kappa usando un clasificador LDA

(b) Matriz de confusión del caso con peor Kappa usando un clasificador SOM

	Aciertos	Falsos positivos	No detección
LDA	50,27 %	64,22 %	35,14 %
SOM	61,41 %	24,35 %	52,93 %

(c) Caso peor Kappa

Cuadro 9: Descriptivo del caso peor en la clasificación de todas las series

7.9. Discusión

Observando los resultados obtenidos en ambos análisis, se puede concluir que SOM ofrecen mejores resultados en cuando a porcentaje de acierto así como falsos positivos, aunque detecta la imaginación de movimiento del sujeto en menor porcentaje que el clasificador LDA. Explicando estos resultados con un ejemplo, si se implementara un sistema BCI con SOM que controlara el encendido o apagado de las luces de una casa domótica, éstas se encenderían erróneamente en media menos que si el sistema funcionara con LDA, pero en cambio al sistema le costaría más detectar la intención del sujeto de encender las luces más que con LDA. Estos parámetros hay que tenerlos en cuenta en el sistema y valorar qué es más crítico, si los falsos positivos o la no detección.

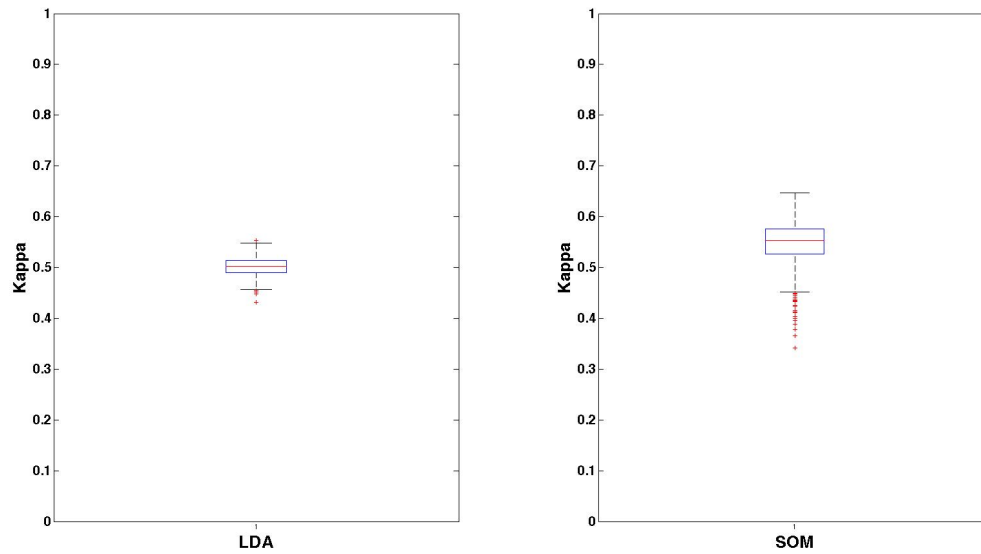


Figura 7.30: Diagramas de caja de las distribuciones de Kappas referentes a todas las series

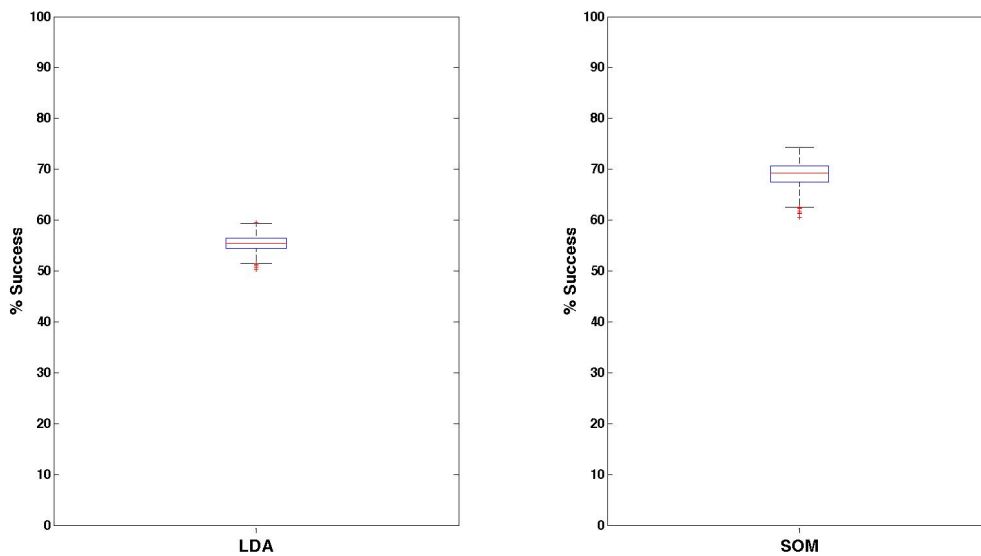


Figura 7.31: Diagramas de caja de las distribuciones de % de acierto referentes a todas las series

7.10. Tabla de tiempos

Las tareas y tiempos referentes al desarrollo de esta fase del proyecto fueron:

<i>Tarea</i>	<i>Horas</i>
Estado del arte EEG/BCI	200
Análisis	80
Diseño	50
Implementación	200
Pruebas	40
Resultados	20
Total	590

Cuadro 10: Tabla de tiempos de la fase de control con EEG

8. Conclusiones y trabajo futuro

En este proyecto final de carrera se alcanzaron satisfactoriamente los siguientes hitos: (1) construcción de un brazo robótico humanoide; (2) implementación de un sistema capaz de detectar mediante la señal electromiográfica (EMG) del antebrazo si la mano del sujeto estaba abierta o cerrada, y controlar mediante ello el brazo robótico; (3) Evaluación del control del brazo robótico mediante análisis de señal electroencefalográfica (EEG) de 103 sujetos; (4) Autofinanciación del proyecto.

El bloque de desarrollo que más trabajo acumuló fue el de evaluación del control mediante EEG, puesto que no hay consenso en la comunidad científica respecto a qué métodos son los más adecuados, y hubo que formar un criterio propio para tomar decisiones en la fase de diseño. Aun así, teniendo en cuenta que se emplearon datos de 103 sujetos, los resultados están por encima de la media del estado del arte, hito adicional conseguido que da pie a que se continúe con el desarrollo de nuevos análisis en esta línea. La inversión realizada en este proyecto se estima en unos 350€. Con la consecución de estos objetivos, se consiguió establecer la base de un entorno de experimentación que servirá como estructura principal para el desarrollo e investigación de futuros proyectos relacionados con el tratamiento de señales bioeléctricas y control.

Dado que se pretende que exista una continuidad del proyecto, y con la experiencia adquirida en este trabajo, se proponen a continuación una serie de recomendaciones para futuros desarrollos:

Brazo robótico

Continuar con la construcción del resto del robot InMoov para poder incorporar más grados de libertad al sistema y poder realizar pruebas relacionadas con otros músculos o miembros superiores completos.

Control por EMG

Implementar un clasificador más refinado que intente decodificar más movimientos de la mano o de los dedos. Para ello se recomienda leer el trabajo de Scott Saponas et al. [37]. También será necesario ampliar el número de canales de registro de EMG. Como se requiere un tratamiento de señal más sofisticado, y la placa que se ha utilizado en este trabajo contenía unas etapas de rectificado y suavizado, se recomienda cambiar a la placa Olimex SHIELD-EKG-EMG (<https://www.olimex.com/Products/Duino/Shields/SHIELD-EKG-EMG/>) que está diseñada como módulo de Arduino y se pueden apilar hasta 6 canales EMG.

En la línea de intentar controlar un exobrazo con señal EMG, se recomienda explorar la decodificación de los músculos involucrados en los siguientes movimientos:

- Elevación/depresión y Retracción/protracción de la articulación escapulo-torácica.
- Flexión/extensión, aducción/abducción y rotación interna/externa de la cabeza Glenohumeral.
- Flexión/extensión y pronación/supinación del codo.
- Flexión/extensión y desviación radial/ulnar de la muñeca.

Control por EEG

El sistema se ha diseñado de forma modular expresamente para probar nuevos métodos en cualquiera de las fases de las que consta la pipeline del sistema (filtrado, extracción de características, clasificación, validación). Se recomienda explorar nuevos:

- Algoritmos de filtrado de artefactos.
- Métodos de extracción de características de la señal cerebral que puedan ser útiles para clasificar tareas motoras.
- Algoritmos de clasificación.
- Métodos de validación.

También se recomienda adaptar el sistema para que pueda ser usado a tiempo real con un equipo de adquisición de señal EEG.

Referencias

- [1] G. A. Donnan, M. Fisher, M. Macleod, and S. M. Davis, "Stroke," *The Lancet*, vol. 371, pp. 1612–1623, May 2008.
- [2] R. W. Teasell and L. Kalra, "What's New in Stroke Rehabilitation Back to Basics," *Stroke*, vol. 36, pp. 215–217, Jan. 2005.
- [3] R. W. Teasell and L. Kalra, "What's New in Stroke Rehabilitation," *Stroke*, vol. 35, pp. 383–385, Jan. 2004.
- [4] V. S. Huang and J. W. Krakauer, "Robotic neurorehabilitation: a computational motor learning perspective," *Journal of NeuroEngineering and Rehabilitation*, vol. 6, p. 5, Feb. 2009.
- [5] K. Laver, S. George, J. Ratcliffe, and M. Crotty, "Virtual reality stroke rehabilitation – hype or hope?," *Australian Occupational Therapy Journal*, vol. 58, no. 3, pp. 215–219, 2011.
- [6] M. A. Oskoei and H. Hu, "Myoelectric control systems—A survey," *Biomedical Signal Processing and Control*, vol. 2, no. 4, pp. 275 – 294, 2007.
- [7] B. Peerdeman, D. Boere, H. Witteveen, R. H. in 't Veld, H. Hermens, S. Stramigioli, H. Rietman, P. Veltink, and S. Misra, "Myoelectric forearm prostheses: state of the art from a user-centered perspective.," *Journal of rehabilitation research and development*, vol. 48, no. 6, pp. 719–737, 2011.
- [8] A. Phinyomark, F. Quaine, S. Charbonnier, C. Serviere, F. Tarpin-Bernard, and Y. Laurillau, "EMG feature evaluation for improving myoelectric pattern recognition robustness," *Expert Systems with Applications*, vol. 40, pp. 4832–4840, Sept. 2013.
- [9] E. Boschmann, P. Kaufmann, M. Platzner, and M. Winkler, *Towards Multi-movement Hand Prostheses: Combining Adaptive Classification with High Precision Sockets*.
- [10] L. Chen, Y. Geng, and G. Li, "Effect of upper-limb positions on motion pattern recognition using electromyography," in *Image and Signal Processing (CISP), 2011 4th International Congress on*, vol. 1, pp. 139–142, Oct. 2011.
- [11] A. Fougner, E. Scheme, A. Chan, K. Englehart, and O. Stavdahl, "Resolving the Limb Position Effect in Myoelectric Pattern Recognition," *Neural Systems and Rehabilitation Engineering, IEEE Transactions on*, vol. 19, pp. 644–651, Dec. 2011.
- [12] M. A. L. Nicolelis, "Actions from thoughts," *Nature*, vol. 409, no. 6818, pp. 403–407, 2001.
- [13] G. Muller-Putz, R. Leeb, M. Tangermann, J. Hohne, A. Kubler, F. Cincotti, D. Mattia, R. Rupp, K.-R. Muller, and J. Del R Millan, "Towards Noninvasive Hybrid Brain-Computer Interfaces: Framework, Practice, Clinical Application, and Beyond," *Proceedings of the IEEE*, vol. 103, pp. 926–943, June 2015.

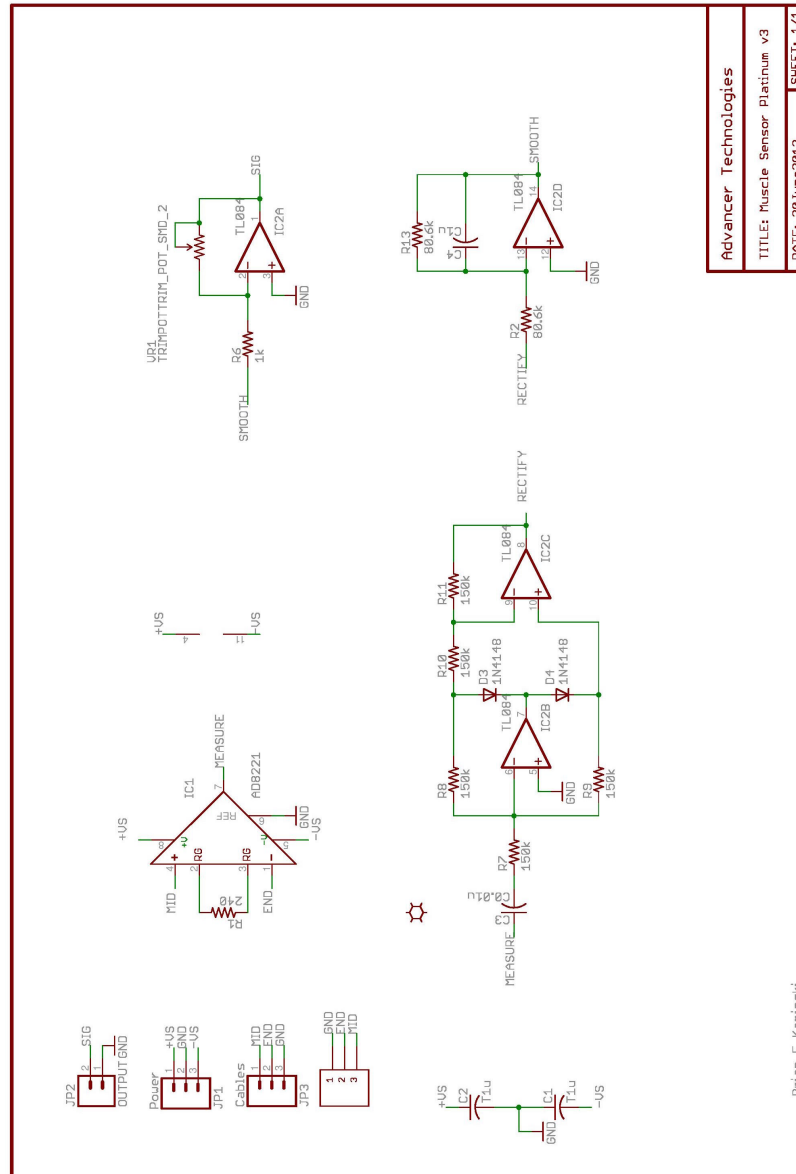
- [14] L. Sörnmo and P. Laguna, “Chapter 3 - EEG signal processing,” in *Bioelectrical Signal Processing in Cardiac and Neurological Applications* (L. S. Laguna, ed.), Biomedical Engineering, pp. 55–179, Burlington: Academic Press, 2005.
- [15] L. Sörnmo and P. Laguna, “Chapter 2 - The electroencephalogram— A brief background,” in *Bioelectrical Signal Processing in Cardiac and Neurological Applications* (L. S. Laguna, ed.), Biomedical Engineering, pp. 25–53, Burlington: Academic Press, 2005.
- [16] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, “PhysioBank, PhysioToolkit, and PhysioNet Components of a New Research Resource for Complex Physiologic Signals,” *Circulation*, vol. 101, pp. e215–e220, June 2000.
- [17] G. Schalk, D. McFarland, T. Hinterberger, N. Birbaumer, and J. Wolpaw, “BCI2000: a general-purpose brain-computer interface (BCI) system,” *IEEE Transactions on Biomedical Engineering*, vol. 51, pp. 1034–1043, June 2004.
- [18] V. Jurcak, D. Tsuzuki, and I. Dan, “10/20, 10/10, and 10/5 systems revisited: Their validity as relative head-surface-based positioning systems,” *NeuroImage*, vol. 34, pp. 1600–1611, Feb. 2007.
- [19] P. Comon, “Independent component analysis, A new concept?,” *Signal Processing*, vol. 36, pp. 287–314, Apr. 1994.
- [20] A. Hyvärinen and E. Oja, “Independent component analysis: algorithms and applications,” *Neural Networks*, vol. 13, pp. 411–430, June 2000.
- [21] “Independent Component Analysis as a tool to eliminate artifacts in EEG: a quantitative study - Journal of Clinical Neurophysiology.”
- [22] A. Hyvarinen, “Fast and robust fixed-point algorithms for independent component analysis,” *IEEE Transactions on Neural Networks*, vol. 10, pp. 626–634, May 1999.
- [23] M. Chaumon, D. V. M. Bishop, and N. A. Busch, “A practical guide to the selection of independent components of the electroencephalogram for artifact correction,” *Journal of Neuroscience Methods*, vol. 250, pp. 47–63, July 2015.
- [24] J. Ma, P. Tao, S. Bayram, and V. Svetnik, “Muscle artifacts in multichannel EEG: Characteristics and reduction,” *Clinical Neurophysiology*, vol. 123, pp. 1676–1686, Aug. 2012.
- [25] I. I. Goncharova, D. J. McFarland, T. M. Vaughan, and J. R. Wolpaw, “EMG contamination of EEG: spectral and topographical characteristics,” *Clinical Neurophysiology*, vol. 114, pp. 1580–1593, Sept. 2003.
- [26] G. Pfurtscheller and F. H. Lopes da Silva, “Event-related EEG/MEG synchronization and desynchronization: basic principles,” *Clinical Neurophysiology*, vol. 110, pp. 1842–1857, Nov. 1999.

- [27] G. Pfurtscheller and A. Aranibar, "Evaluation of event-related desynchronization (ERD) preceding and following voluntary self-paced movement," *Electroencephalography and Clinical Neurophysiology*, vol. 46, no. 2, pp. 138 – 146, 1979.
- [28] G. Pfurtscheller and A. Berghold, "Patterns of cortical activation during planning of voluntary movement," *Electroencephalography and Clinical Neurophysiology*, vol. 72, no. 3, pp. 250 – 258, 1989.
- [29] P. Derambure, L. Defebvre, K. Dujardin, J. L. Bourriez, J. M. Jacquesson, A. Destee, and J. D. Guieu, "Effect of aging on the spatio-temporal pattern of event-related desynchronization during a voluntary movement," *Electroencephalography and Clinical Neurophysiology/Evoked Potentials Section*, vol. 89, no. 3, pp. 197 – 203, 1993.
- [30] C. Toro, G. Deuschl, R. Thatcher, S. Sato, C. Kufta, and M. Hallett, "Event-related desynchronization and movement-related cortical potentials on the {ECoG} and {EEG}," *Electroencephalography and Clinical Neurophysiology/Evoked Potentials Section*, vol. 93, no. 5, pp. 380 – 389, 1994.
- [31] A. S. Jr and G. Pfurtscheller, "Event-related desynchronisation of central beta-rhythms during brisk and slow self-paced finger movements of dominant and nondominant hand," *Cognitive Brain Research*, vol. 4, no. 3, pp. 171 – 183, 1996.
- [32] L. Leocani, C. Toro, P. Manganotti, P. Zhuang, and M. Hallett, "Event-related coherence and event-related desynchronization/synchronization in the 10 Hz and 20 Hz {EEG} during self-paced movements," *Electroencephalography and Clinical Neurophysiology/Evoked Potentials Section*, vol. 104, no. 3, pp. 199 – 206, 1997.
- [33] G. Pfurtscheller and C. Neuper, "Motor imagery activates primary sensorimotor area in humans," *Neuroscience Letters*, vol. 239, no. 2–3, pp. 65 – 68, 1997.
- [34] T. Kohonen, "The self-organizing map," *Proceedings of the IEEE*, vol. 78, pp. 1464–1480, Sept. 1990.
- [35] J. Cohen, "A Coefficient of Agreement for Nominal Scales," *Educational and Psychological Measurement*, vol. 20, pp. 37–46, Apr. 1960.
- [36] R. L. Brennan and D. J. Prediger, "Coefficient Kappa: Some Uses, Misuses, and Alternatives," *Educational and Psychological Measurement*, vol. 41, pp. 687–699, Oct. 1981.
- [37] T. S. Saponas, D. S. Tan, D. Morris, R. Balakrishnan, J. Turner, and J. A. Landay, "Enabling always-available input with muscle-computer interfaces," in *Proceedings of the 22nd annual ACM symposium on User interface software and technology*, pp. 167–176, 2009.

9. Anexo I: ESQUEMA PLACA EMG

Esquema en pdf y manual disponibles en:

<http://www.advancertechnologies.com/p/muscle-sensor-v3.html>



2/4/2013 1:03:11 PM H:\EMGMuscle Sensor Platinum\v3\Muscle Sensor Platinum v3.3.sch (Sheet: 1/1)

Figura 9.32: Esquema de la placa de adquisición de señal EMG

10. Anexo II: ANÁLISIS DE COMPONENTES INDEPENDIENTES

[Nota a pie de página]²

10.1. Definición de ICA

Asumiendo que las n señales que observamos, x_1, x_2, \dots, x_n , son una combinación lineal de n fuentes independientes

$$x_j(t) = a_{j1}s_1(t) + a_{j2}s_2(t) + \dots + a_{jn}s_n(t) \quad (9)$$

o, usando notación matricial, $x = As$, nuestro objetivo es encontrar la matriz de desmezclado W tal que $s = Wx$. Además, asumimos que:

- Tanto las señales mezcladas como las fuentes tienen media cero (i.e. $E[x_i] = E[s_j] = 0, \forall i, j$). Si no es así, simplemente sustraemos la media.
- Las fuentes tienen distribuciones no gaussianas.
- La matriz de mezclado es cuadrada, i.e., hay tantas fuentes como señales mezcladas.

10.2. Ambigüedades de ICA

La varianza de las componentes independientes no puede ser determinada:

- Dado que las fuentes, s , y la matriz de mezclado, A , no están determinadas, cualquier factor que multiplique a s , incluyendo un cambio de signo, puede ser contrarrestado por los coeficientes en A

$$x_j(t) = (ka_{j1})s_1(t) + (ka_{j2})s_2(t) = a_{j1}(ks_1(t)) + a_{j2}(ks_2(t)) \quad (10)$$

- Para resolver esta ambigüedad, es asumido que las fuentes tienen varianza unidad

$$E[s_i^2] = 1 \quad (11)$$

El orden de las componentes independientes no puede ser determinado:

- Dado que tanto s como A no son conocidos, cualquier permutación de los términos de mezclado puede dar el mismo resultado.

²Esta sección ha sido traducida y adaptada de los apuntes del curso "Pattern Analysis" de Ricardo Gutierrez-Osuna, disponibles en http://research.cs.tamu.edu/prism/lectures/pr/pr_127.pdf

10.3. Independencia vs no correlación

¿Qué es independencia?

- Dos variables aleatorias y_1 e y_2 se dice que son independientes si el conocimiento del valor de una no aporta ninguna información respecto al valor de la otra y viceversa

$$p(y_1|y_2) = p(y_1) \Leftrightarrow p(y_1, y_2) = p(y_1)p(y_2) \quad (12)$$

¿Qué es no correlación?

- Dos variables aleatorias y_1 e y_2 se dice que no están correladas si su covarianza es cero

$$E[y_1^2 y_2^2] = 0 \quad (13)$$

Equivalencias

- Independencia implica no correlación
- La no correlación no implica independencia (salvo que las dos variables aleatorias sean gaussianas, en ese caso independencia y no correlación son equivalentes)

10.4. Independencia vs No Gaussianidad

Como hemos visto anteriormente, una condición necesaria para que ICA funcione es que las señales sean no gaussianas

- Si no se cumple, ICA no puede resolver las direcciones de las componentes independientes debido a simetrías

Ahora vamos a ver que encontrar las componentes independientes es equivalente a encontrar las direcciones de máxima no-gaussianidad.

- Para simplificar, vamos a asumir que las fuentes tienen idénticas distribuciones
- Nuestro objetivo es encontrar el vector w tal que $y = w^T x$ sea igual a una de las fuentes s
- Realizamos el cambio de variable $z = A^T w$,
 - Esto nos lleva a que $y = w^T x = w^T A s = z^T s$
 - Por consiguiente, y es una combinación lineal de las fuentes s
- De acuerdo con el Teorema Central del Límite, la señal y es más gaussiana que las fuentes s , ya que es una combinación lineal de ellas, y es lo menos gaussiana cuando es igual a una de las fuentes.
- Por lo tanto, el w óptimo es el vector que maximiza la no-gaussianidad de $w^T x$, dado que ello hará que y sea igual a una de las fuentes.
- El truco pasa a ser cómo medir la no-gaussianidad

- La entropía de una variable puede ser vista como medida de aleatoriedad
- Para una variable discreta, la entropía $H(Y)$ es definida como

$$H(Y) = - \sum_i P(Y = a_i) \log(P(Y = a_i)) \quad (14)$$

- Mientras que para una variable continua, la entropía diferencial es

$$H(Y) = - \int p(y) \log(p(y)) dy \quad (15)$$

- Para obtener una medida de no-gaussianidad, uno puede tomar de medida la diferencia de entropía relativa a una gaussiana; esto es conocido como neguentropía

$$J(y) = H(y_G) - H(y), \quad (16)$$

donde y_G es una variable gaussiana con la misma varianza que y

- Notar que $J(y)$ siempre es no negativa, y solo igual a cero para una gaussiana.
- Puesto que la estimación de la neguentropía es difícil, típicamente se usan aproximaciones propuestas por Hyvarinen, que tienen la forma

$$J(y) \propto [E(G(y)) - E(G(v))]^2, \quad (17)$$

donde v es una variable Gaussiana $N(0,1)$, y se asume que tiene media cero, y G es una función no cuadrática

- Otras elecciones de G han dado buenos resultados, incluyendo

$$G_1(u) = \frac{1}{a_1} \log(\cosh(a_1 u)), 1 \leq a_1 \leq 2 \quad (18)$$

$$G_2(u) = -e^{-\frac{u^2}{2}} \quad (19)$$

10.5. Preprocesado de ICA

Se pueden mejorar las condiciones iniciales de ICA si los datos son preprocesados antes del análisis:

Centrado

- este paso consiste en sustraer la media del vector observado

$$x' = x - E[x] \quad (20)$$

- La media del vector puede ser añadida a las fuentes estimadas posteriormente

$$s = s' + A_{-1}E[x] \quad (21)$$

Blanqueado

- consiste en aplicar una transformación lineal a los datos para decorrelar sus componentes y que tengan varianza unidad

$$\tilde{x} = Wx \Rightarrow E[\tilde{x}\tilde{x}] = I \quad (22)$$

- esto se puede conseguir a través de las componentes principales

$$\tilde{x} = ED^{-\frac{1}{2}}E^T x, \quad (23)$$

donde las columnas de E y la diagonal de D son los vectores propios y valores propios de $E[xx^T]$ respectivamente

$$E[xx^T] = EDE^T \quad (24)$$

10.6. Algoritmo FastICA para una unidad

FastICA es un método muy eficiente de maximización de medidas de no-gaussianidad mencionadas anteriormente. A partir de ahora, asumimos que los datos han sido centrados y blanqueados. Comenzamos con el problema de estimar una única fuente y después lo extendemos a varias fuentes.

El objetivo es encontrar un vector de pesos w que maximice la neguentropía estimada

$$J(w^T x) \propto [E(G(w^T x)) - E(G(v))]^2, \quad (25)$$

notar que el máximo de $J(w^T x)$ ocurre dado un cierto valor óptimo de $E(G(w^T x))$, dado que la segunda parte es independiente de w .

De acuerdo con las condiciones de Kuhn-Tucker, el valor óptimo de $E(G(w^T x))$, bajo la restricción $E(G(w^T x))^2 = \|w\|^2 = 1$, ocurre en el punto donde

$$F(w) = E[xg(w^T x)] - \beta w = 0 \quad (26)$$

$$\text{donde } g(u) = \frac{dG(u)}{du} \quad (27)$$

La restricción $E(G(w^T x))^2 = \|w\|^2 = 1$ se debe a que la varianza de $w^T x$ tiene que ser igual a la unidad (por diseño): si los datos han sido blanqueados al inicio, entonces el módulo de w tiene que ser igual a 1.

El problema puede ser resuelto como una aproximación por el método de Newton. Para encontrar un cero de $f(x)$, aplicar la iteración

$$x_n = x_n - \frac{f(x_n)}{f'(x_n)} \quad (28)$$

Calculando el jacobiano de $F(w)$ tenemos

$$JF(X) = Exx^T g'(w^T x) - \beta I \quad (29)$$

para simplificar la inversión de la matriz, aproximamos el primer término de la expresión,

$$E[xx^T g(w^T x)] \approx E[xx^T]E[g'(w^T x)] = E[g'(w^T x)]I \quad (30)$$

dado que $E[g'(w^T x)]$ es un escalar, tenemos que el Jacobiano es diagonal, lo que simplifica la inversión. Por tanto, la iteración (aproximación) de Newton es

$$w^+ = w - \frac{(E[xg(w^T x)] - \beta w)}{(E[g'(w^T x)] - \beta)} \quad (31)$$

el algoritmo puede ser simplificado aún más, multiplicando a ambos lados por $\beta - E[g'(w^T x)]$, lo que nos da el algoritmo iterativo de FastICA

1. Elegir un vector de pesos inicial (e.g. aleatorio)
2. Sea $w^+ = E[xg(w^T x)] - E[g'(w^T x)]w$
3. Sea $w = \frac{w^+}{\|w^+\|}$
4. Si no converge, volver a 2

10.7. Algoritmo FastICA para varias unidades

Para estimar varias componentes independientes, lanzamos el algoritmo FastICA con varias unidades w_1, w_2, \dots, w_n . Para prevenir que varios de estos vectores converjan a la misma solución, decorrelamos las salidas $w_1^T x, w_2^T x, \dots, w_n^T x$ en cada iteración.

Esto puede hacerse usando el esquema de deflación basado en Gram-Schmidt:

- Estimamos cada componente una a una)
- Con p componentes estimadas w_1, w_2, \dots, w_p lanzamos una iteración de ICA de una unidad para w_{p+1}
- Después de cada iteración, sustraemos de w_{p+1} sus proyecciones $(w_{p+1}^T w_j)w_j$ del vector anterior w_j
- Después, renormalizamos w_{p+1}

a) Sea $w_{p+1} = w_{p+1} - \sum_{j=1}^p w_{p+1}^T w_j w_j$

b) Sea $w_{p+1} = \frac{w_{p+1}}{\sqrt{w_{p+1}^T w_{p+1}}}$

11. Anexo III: DIAGRAMA DE GANTT

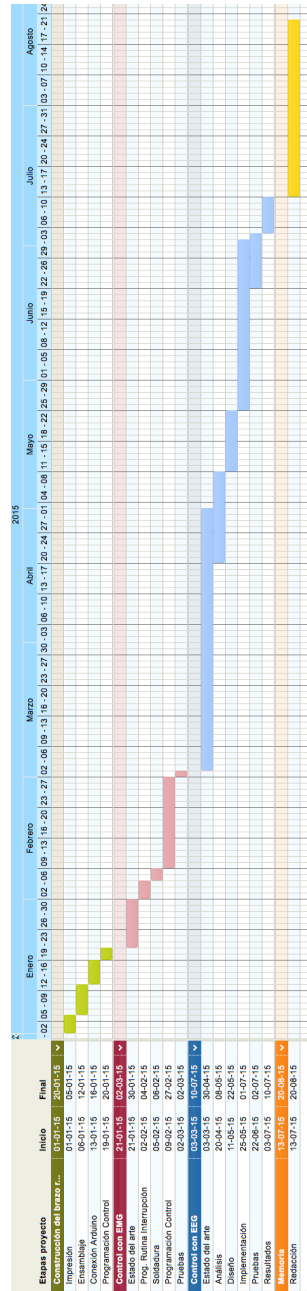


Figura 11.33: Diagrama de Gantt del proyecto

12. Anexo IV: TABLA TOTAL DE TIEMPOS

<i>Tarea</i>	<i>Horas</i>
Brazo robótico	
Impresión de las piezas	20
Ensamblaje	15
Conexión con Arduino	16
Programación del control	10
EMG	
Estado del arte	40
Programación de la rutina de interrupción de reloj en Arduino para el muestreo a 500Hz	10
Soldadura de pines y cables para integrar la placa en Arduino	5
Programación del control en Matlab	20
Pruebas	2
EEG	
Estado del arte EEG/BCI	200
Análisis	80
Diseño	50
Implementación	200
Pruebas	40
Resultados	20
Memoria	
Redacción	100
Total	828

Cuadro 11: Tabla total de tiempos del proyecto