



UNIVERSIDAD DE MÁLAGA



Graduada en Ingeniería de la Salud

Control de un brazo robótico mediante el procesamiento de señales electromiográficas

Control of a robotic arm through the processing of
electromyographic signals

Realizado por
Raquel Martínez Pérez

Tutorizado por
Antonio Jesús Bandera Rubio
Alberto José Tudela Roldán

Departamento
Tecnología Electrónica

MÁLAGA, septiembre de 2025



UNIVERSIDAD
DE MÁLAGA



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INFORMÁTICA
GRADUADA EN INGENIERÍA DE LA SALUD

**Control de un brazo robótico mediante el procesamiento
de señales electromiográficas**

**Control of a robotic arm through the processing of
electromyographic signals**

Realizado por

Raquel Martínez Pérez

Tutorizado por

Antonio Jesús Bandera Rubio

Alberto José Tudela Roldán

Departamento

Tecnología Electrónica

UNIVERSIDAD DE MÁLAGA

MÁLAGA, SEPTIEMBRE DE 2025

Fecha defensa: Septiembre de 2025

Resumen

Alrededor de 1,5 de cada 1.000 personas en el mundo viven con la pérdida de alguna extremidad, causada principalmente por accidentes o enfermedades como cáncer, diabetes u osteomielitis. Cerca del 30% de estas amputaciones afectan al miembro superior, lo que supone unos 3 millones de personas a nivel global. Esta condición conlleva una importante pérdida de autonomía y, aunque existen prótesis para mitigarla, muchas presentan un control limitado que dificulta su adaptación y uso.

En este Trabajo de Fin de Grado se ha diseñado un sistema de control de un brazo robótico a partir de señales electromiográficas, capaces de interpretar la actividad muscular y transformarla en movimientos útiles, funcionando como un prototipo de prótesis. El proceso incluyó la adquisición de señales mediante electrodos en el antebrazo, su procesamiento mediante centrado en cero para eliminar la componente continua, filtrado Notch para atenuar el ruido de la red eléctrica y filtrado paso banda para reducir ruido y artefactos de movimiento, así como la extracción de características en el dominio temporal y frecuencial. Con estos datos se entrenó un clasificador basado en el algoritmo Ensemble Subspace Discriminant, cuya salida permitió enviar órdenes al brazo robótico y ejecutar los movimientos correspondientes.

El sistema mostró una gran exactitud en el reconocimiento de gestos, lo que refuerza su viabilidad y lo convierte en una base prometedora para el desarrollo de prótesis mioeléctricas más precisas, naturales y fáciles de usar.

Palabras clave: electromiografía, prótesis mioeléctrica, brazo robótico, clasificación de señales EMG, detección de movimientos.

Abstract

Approximately 1.5 out of every 1,000 individuals worldwide live with the loss of a limb, primarily due to accidents or medical conditions such as cancer, diabetes, or osteomyelitis. Nearly 30% of these amputations affect the upper limb, representing an estimated 3 million people globally. This condition entails a significant loss of autonomy and, although prostheses exist to mitigate this limitation, many exhibit restricted control that hinders adaptation and daily use.

In this project, a control system for a robotic arm was designed based on electromyographic (EMG) signals, capable of interpreting muscular activity and transforming it into functional movements, serving as a prosthetic prototype. The process involved signal acquisition through electrodes placed on the forearm, followed by preprocessing steps including zero-centering to remove the DC component, notch filtering to suppress powerline interference, and band-pass filtering to reduce noise and motion artifacts. Subsequently, feature extraction was performed in both the time and frequency domains. These features were used to train a classifier based on the Ensemble Subspace Discriminant algorithm, whose outputs enabled the robotic arm to execute the corresponding movements.

The system demonstrated high accuracy in gesture recognition, reinforcing its feasibility and establishing a promising foundation for the development of more precise, natural, and user-friendly myoelectric prostheses.

Keywords: electromyography, myoelectric prosthesis, robotic arm, EMG signal classification, movement detection.

Índice

Índice de figuras

Índice de tablas

Introducción.....	1
1.1 Motivación	1
1.2 Objetivos	2
1.3 Estructura de la memoria	2
1.4 Estado del arte	4
Electromiografía	7
2.1. Introducción a la electromiografía	7
2.2. Fundamento fisiológico del movimiento muscular.....	9
2.3. Características de la señal EMG	12
Anatomía neuromuscular del antebrazo	15
3.1 Compartimento anterior	21
3.2 Compartimento posterior	23
3.3 Compartimento lateral.....	25
3.4 Inervación de los músculos del antebrazo.....	27
Adquisición de señales EMG	29
4.1 Sistema de adquisición	30
4.2 Adquisición de datos	33
Procesamiento de la señal EMG.....	37
5.1 Centrado en cero	38
5.2 Filtro Notch	42
5.3 Filtro paso banda.....	46
Extracción de características.....	49
6.1 Dominio temporal	50
6.2 Dominio frecuencial	52
Clasificador de gestos	55

7.1 Estandarización de los datos	55
7.2 Implementación del clasificador	56
7.3 Comparación de modelos.....	57
7.4 Resultados del entrenamiento	61
Control del brazo robótico	65
8.1 Estructura, control y alimentación del brazo robótico	66
8.2 Restauración y puesta en funcionamiento del brazo.....	67
8.3 Integración del clasificador de gestos.....	69
Conclusiones y líneas futuras.....	73
Bibliografía.....	75
Apéndice A: Código fuente	83
Procesamiento de los datos de entrenamiento.....	83
1. Configuración inicial.....	83
2. Centrado en cero para eliminar la componente continua	84
3. Transformada Rápida de Fourier	84
4. Filtro Notch.....	85
5. Filtro paso banda	86
6. Eliminación de subcarpetas del struct para facilitar el procesamiento	86
7. Extracción de características	87
8. Adición de la columna de clase.....	88
9. Estandarización de características	89
Script principal: adquisición, procesamiento, clasificación y control del brazo	89
Función de lectura de señales	90
Función de procesamiento.....	91
Función de clasificación de gestos.....	94
Función de control del robot.....	94

Índice de figuras

Figura 1: Composición del músculo esquelético.....	9
Figura 2: Unión neuromuscular.....	10
Figura 3: Potencial de acción.	11
Figura 4: La electromiografía, suma de los potenciales de acción de distintas unidades motoras.....	12
Figura 5: Espectro de frecuencias de la señal EMG.	13
Figura 6: Apertura de mano.....	16
Figura 7: Cierre de mano	16
Figura 8: Pronación del antebrazo	16
Figura 9: Supinación del antebrazo	17
Figura 10: Posición neutra	17
Figura 11: Extensión de pulgar.....	17
Figura 12: Extensión-abducción del segundo y tercer dedos	17
Figura 13: Compartimentos del antebrazo.....	18
Figura 14: Músculos del antebrazo (corte transversal).....	20
Figura 15: Músculos del antebrazo (corte frontal).	20
Figura 16: Músculos flexores y pronadores del antebrazo (compartimento anterior).	23
Figura 17: Músculos extensores, braquiorradial y supinador del antebrazo (compartimento posterior y lateral).	26
Figura 18: Canales de adquisición de datos y de referencia de la PCB	30
Figura 19: Sistema de adquisición de señales EMG	30
Figura 20: Placa de circuito impreso del sistema de adquisición	31
Figura 21: Colocación de los electrodos	34
Figura 22: Apertura de la mano (Señal cruda)	35
Figura 23: Apertura de la mano (Señal cuantificada en tensión)	35
Figura 24: Apertura de la mano (Centrado en cero)	38
Figura 25: Cierre de la mano (Centrado en cero).....	39
Figura 26: Pronación (Centrado en cero)	39

Figura 27: Supinación (Centrado en cero)	40
Figura 28: Posición neutra (Centrado en cero)	40
Figura 29: Extensión de pulgar (Centrado en cero)	41
Figura 30: Extensión segundo y tercer dedos (Centrado en cero).....	41
Figura 31: Comparación de la FFT en apertura y reposo	43
Figura 32: Respuesta en frecuencia del filtro Notch, centrado en la frecuencia normalizada	45
Figura 33: Comparación de la señal tras aplicar el filtro Notch.....	45
Figura 34: Comparativa de las FFT antes y después de aplicar el filtro Notch	46
Figura 35: Comparativa de la señal antes y después del filtro paso banda	47
Figura 36: Tabla de características temporales y frecuenciales extraídas por canal y movimiento.....	54
Figura 37: Configuración inicial de parámetros en la aplicación Classification Learner	57
Figura 38: Comparación de las curvas ROC de los diferentes modelos	59
Figura 39: Comparación de la exactitud de los modelos.....	60
Figura 40: Comparación de la tasa de error de los modelos	60
Figura 41: Comparación del coste computacional de los modelos	60
Figura 42: Resultados del entrenamiento del Ensemble Subspace Discriminant	62
Figura 43: Matriz de confusión, TPR y FNR del modelo elegido	63
Figura 44: Robot completo	66
Figura 45: Componentes del brazo	67
Figura 46: Montaje de la nueva pieza impresa de la muñeca	68
Figura 47: Brazo completo y restaurado	68
Figura 48: Script principal para el control del brazo	71
Figura 49: Respuesta del brazo a la extensión del pulgar obtenida como salida del clasificador	71

Índice de tablas

Tabla 1: Movimientos realizados durante el estudio	17
Tabla 2: Músculos del antebrazo agrupados por compartimento y función.	19
Tabla 3: Inervación de los músculos del antebrazo	28
Tabla 4: Comparación de la exactitud, tasa de error y coste computacional de los diferentes modelos	60
Tabla 5: Comparación de los resultados de las pruebas de los clasificadores	61
Tabla 6: Codificación del movimiento del robot en función del gesto realizado	70

1

Introducción

1.1 Motivación

La pérdida de una extremidad superior constituye un acontecimiento con un impacto significativo a nivel funcional, psicológico y social en la vida de la persona afectada. A partir de ese momento, actividades básicas de la vida diaria como alimentarse, vestirse o asearse pueden llegar a convertirse en tareas extremadamente complejas, requiriendo en muchos casos la asistencia de una tercera persona para ser realizadas.

No obstante, esta ayuda no siempre está disponible. Algunas personas no disponen de una red de apoyo familiar cercana, mientras que otras carecen de los recursos económicos necesarios para contratar servicios de asistencia personal. Estas circunstancias evidencian el grado de vulnerabilidad y dependencia al que puede verse sometido un individuo cuando su capacidad funcional se ve seriamente comprometida.

En este contexto, el presente Trabajo de Fin de Grado plantea el desarrollo de un sistema de control para un brazo robótico que permita ejecutar, con precisión, los movimientos que el usuario desea realizar, actuando como una prolongación natural del miembro perdido. El objetivo principal es facilitar que la persona amputada recupere, en la

medida de lo posible, su autonomía en la realización de actividades cotidianas, minimizando así su dependencia de terceros y mejorando su calidad de vida.

1.2 Objetivos

El objetivo principal de este Trabajo de Fin de Grado es el control del movimiento de un brazo robótico mediante el tratamiento de señales electromiográficas. Para ello, se implementará un sistema capaz de capturar y clasificar las señales procedentes de la actividad eléctrica de los músculos del antebrazo según los movimientos realizados por el sujeto.

Para alcanzar dicho objetivo, se llevarán a cabo los siguientes procedimientos:

- Obtención de las señales electromiográficas mediante un sistema de captura basado en la disposición de electrodos de superficie sobre la piel y un front-end analógico de ocho canales.
- Procesamiento de las señales EMG mediante diferentes filtros para minimizar los posibles ruidos.
- Extracción de características de las señales para su posterior clasificación. Se extraerán características en el dominio temporal y frecuencial que servirán como base del clasificador.
- Identificación del movimiento. Se desarrollará un software basado en algoritmos de inteligencia artificial para la clasificación de movimientos, tomando como entrada las características temporales y frecuenciales de las señales procesadas.
- Transmisión del movimiento al brazo robótico. Se desarrollará un software para transmitir el movimiento detectado al brazo robótico y controlar su movimiento.

1.3 Estructura de la memoria

Este trabajo está estructurado en nueve capítulos, organizados en distintos apartados que permiten seguir de manera ordenada y coherente el desarrollo del proyecto. A continuación, se describen los capítulos que conforman la memoria:

1. **Introducción:** en este capítulo se exponen la motivación y los objetivos del presente trabajo. Asimismo, se presenta la organización de la memoria en capítulos, se describen las tecnologías actuales consideradas en este campo y se detalla la metodología que se ha empleado para su desarrollo.

2. **Electromiografía:** describe los fundamentos de la electromiografía, sus modalidades y las características fisiológicas y técnicas de la señal EMG.
3. **Anatomía neuromuscular del antebrazo:** detalla la estructura muscular, compartimentos y nervios del antebrazo, relacionando cada grupo con los gestos analizados. Este conocimiento permite la correcta colocación de electrodos y la interpretación de las señales.
4. **Adquisición de señales EMG:** explica la metodología de adquisición diseñada, incluyendo el hardware, definición de parámetros y colocación de los electrodos.
5. **Procesamiento de la señal EMG:** aborda las técnicas de procesamiento y filtrado aplicadas a las señales: centrado en cero, filtros notch y paso banda. Estas etapas garantizan la limpieza y fiabilidad de las señales antes del análisis.
6. **Extracción de características:** expone las métricas calculadas en el dominio temporal y frecuencial para describir las señales EMG, con el objetivo de obtener descriptores relevantes que faciliten la clasificación de los gestos.
7. **Clasificador de gestos:** describe los modelos de aprendizaje automático utilizados para clasificar los gestos a partir de las características extraídas, mostrando la validación, precisión y resultados del clasificador seleccionado.
8. **Control del brazo robótico:** explica la integración del sistema de clasificación con el brazo robótico, detallando la transmisión de órdenes, la ejecución de gestos y las pruebas experimentales realizadas para validar el control.
9. **Conclusiones y líneas futuras:** resume los logros alcanzados en el desarrollo del sistema de reconocimiento de gestos y control del brazo robótico, destacando las limitaciones encontradas y proponiendo líneas de mejora para futuras investigaciones.

1.4 Estado del arte

La electromiografía se ha convertido en una de las técnicas más relevantes para el control de prótesis en los últimos años. A través de esta técnica es posible registrar la actividad eléctrica de los músculos y procesarla para reconocer con precisión los gestos que realiza la persona. De este modo, se establece un puente directo entre la intención motora de la persona y la ejecución de movimientos en un dispositivo protésico, lo que representa un avance clave en la recuperación de autonomía y funcionalidad en personas con amputaciones del miembro superior.

En cuanto a las tecnologías actuales de adquisición, destacan dos enfoques principales. Por un lado, el Myo armband, un brazalete con ocho electrodos secos distribuidos alrededor del antebrazo. Este permite registrar señales musculares de manera no invasiva y transmitirlos de forma inalámbrica, facilitando una mayor comodidad de uso por el paciente y reduciendo así el tiempo de preparación del sistema.

Este novedoso sistema, empleado por Tepe y Demir (2020) en su estudio junto a técnicas de procesado y extracción de características, posibilitó el reconocimiento preciso de múltiples gestos en tiempo real. Así, el dispositivo demostró su potencial como herramienta portátil para el control protésico.

Por otro lado, Ibrra Fuentes et. al (2025) exploraron una alternativa de bajo coste y menor complejidad, empleando solo tres electrodos superficiales en músculos extensores del antebrazo y una unidad de medida inercial (IMU) para capturar movimientos de rotación. A través del procesamiento mediante umbrales de activación para diferenciar los movimientos de apertura y cierre de la mano y el control de giro mediante el IMU, lograron controlar los movimientos de una mano robótica.

Respecto a los algoritmos de clasificación, las técnicas más utilizadas incluyen desde métodos tradicionales de aprendizaje supervisado hasta enfoques de aprendizaje profundo. Los clasificadores clásicos, como máquinas de vectores de soporte (SVM), vecinos más cercanos (KNN) o árboles de decisión, destacan por su bajo coste computacional y su elevada precisión al complementarse con características temporales y frecuenciales. En particular,

SVM es el clasificador que ha demostrado mayor robustez en entornos de tiempo real, como evidencia el trabajo de Tepe y Demir (2022).

En paralelo, los modelos de aprendizaje profundo tales como los algoritmos de redes neuronales (CNN, LSTM, etc.) permiten reconocer gestos directamente de la señal cruda, disminuyendo así el número de técnicas complementarias utilizadas y posibilitando el reconocimiento de gestos más complejos. Sin embargo, este tipo de algoritmos requiere mayores recursos computacionales y bases de datos extensas para obtener una precisión elevada.

En conclusión, la combinación de estas técnicas de adquisición y los distintos algoritmos de clasificación permitirá avanzar hacia sistemas de control de prótesis más precisos, estables y accesibles.

1.5 Metodología

Para el desarrollo de este trabajo se ha seguido la metodología en cascada, la cual divide el proyecto en fases e impide el avance a fases posteriores sin finalizar la anterior. De esta manera, el trabajo se estructura en siete fases.

La primera fase aborda el planteamiento y alcance del proyecto, la segunda comprende la obtención de las señales EMG, la tercera consiste en el procesamiento de las señales obtenidas, aplicando filtros cuando sea necesario para minimizar ruidos. La cuarta fase abarca el proceso de extracción de características en el dominio temporal y frecuencial para entrenar al clasificador de movimientos basado en inteligencia artificial, desarrollado en la quinta fase del proyecto y que permitirá la identificación de movimientos. La sexta fase se centra en la transmisión del movimiento detectado en la fase anterior al robot para controlar su movimiento y, finalmente, la séptima fase finaliza con la redacción de la memoria del trabajo.

Esta metodología permite el desarrollo de un proyecto completo y coherente, ya que las fases posteriores del trabajo requieren de datos o resultados de las fases anteriores. Por tanto, así se asegura obtener los resultados deseados para poder continuar con el trabajo.

2

Electromiografía

En esta sección se describirá el procedimiento clínico en el que se basa el presente Trabajo de Fin de Grado para la toma de las señales procedentes del músculo: la electromiografía.

2.1. Introducción a la electromiografía

La electromiografía es la técnica que permite registrar la actividad eléctrica del músculo esquelético en respuesta a la estimulación neuronal. Tradicionalmente se emplea en el ámbito clínico como herramienta diagnóstica para evaluar la función muscular y neurológica, y detectar patologías como la distrofia muscular o la esclerosis lateral amiotrófica (Electromiografía - Mayo Clinic, s. f.). Sin embargo, en este trabajo se utilizará para capturar señales musculares procedentes del antebrazo con el objetivo de controlar un brazo robótico que simule una prótesis mioeléctrica.

Existen dos métodos para realizar una electromiografía (EMG): la EMG intramuscular y la EMG de superficie. La primera se lleva a cabo mediante la inserción de una aguja hipodérmica a través de la piel del paciente hasta alcanzar el músculo. Se trata de un método

invasivo, pero permite registrar la actividad de músculos aislados y profundos que, de otro modo, no serían accesibles. Además, según Guzmán-Muñoz y Méndez-Rebolledo (2018), ofrece mejores resultados que la EMG superficial, ya que, al estar en contacto directo con el músculo, se reduce el ruido generado por la interferencia de señales eléctricas de músculos cercanos, fenómeno conocido como *crosstalk*.

No obstante, la contracción muscular puede verse alterada debido a la molestia provocada por la aguja, lo cual dificulta la adquisición precisa de señales. Asimismo, es un método poco reproducible, dado que resulta complejo insertar la aguja exactamente en el mismo punto en mediciones sucesivas, lo que podría implicar el registro de un músculo diferente (Aparicio, 2005).

Por su parte, la EMG superficial consiste en la disposición de electrodos adhesivos sobre la piel del paciente de manera no invasiva. Es una técnica indolora, segura y más cómoda tanto para el sujeto como para el operador, por lo que resulta especialmente adecuada para aplicaciones no clínicas, como es el caso de este trabajo. Además, es un método muy reproducible y que permite un estudio de un conjunto de músculos, en contraposición de su alternativa intramuscular (Aparicio, 2005).

Sin embargo, al igual que esta, presenta ciertas limitaciones. Al situarse los electrodos a una mayor distancia del músculo objetivo, la señal registrada puede verse atenuada debido a la resistencia de los tejidos interpuestos (piel, grasa subcutánea, etc.), lo que disminuye la amplitud y calidad del registro (Guzmán-Muñoz y Méndez-Rebolledo, 2018). Además, existe una mayor susceptibilidad a la captación de señales provenientes de músculos adyacentes (*crosstalk*), lo que puede comprometer la especificidad de la señal. Por otro lado, el contacto entre el electrodo y la piel puede generar artefactos si la impedancia no es adecuada o si existen restos de vello o suciedad.

Para minimizar estos inconvenientes y optimizar la calidad del registro, es fundamental realizar una correcta preparación de la piel. Según el proyecto SENIAM (Surface Electromyography for the Non-Invasive Assessment of Muscles project of the European Union), esta preparación deberá incluir el rasurado del área si fuera necesario y la limpieza con alcohol para reducir la impedancia y mejorar la conductividad eléctrica de la piel (Welcome To SENIAM, s. f.). De esta forma, se maximiza la relación señal/ruido y se obtienen resultados más fiables.

Por tanto, considerando el propósito de este trabajo, se ha decidido emplear la electromiografía superficial como técnica para la obtención de las señales musculares, ya que se pretende analizar de forma global la actividad de la musculatura del antebrazo y no se requiere una precisión elevada en la identificación individual de cada músculo. Además, dado que las señales se recogerán de varios sujetos, este método resulta más adecuado por la facilidad y comodidad que ofrece en la colocación de los electrodos, favoreciendo así la repetibilidad y eficiencia del procedimiento.

2.2. Fundamento fisiológico del movimiento muscular

El músculo esquelético, es decir, aquel que está unido al hueso mediante tendones y que permite el movimiento voluntario del cuerpo, está formado por fibras musculares agrupadas y rodeadas por tejido conectivo o fascia, como se muestra en la Figura 1. Cada conjunto formado por una neurona motora (motoneurona) y todas las fibras musculares que inerva recibe el nombre de unidad motora. La conexión entre la motoneurona y cada fibra muscular se realiza a través de una estructura especializada denominada placa motora, formando la unión neuromuscular, como se aprecia en la Figura 2.

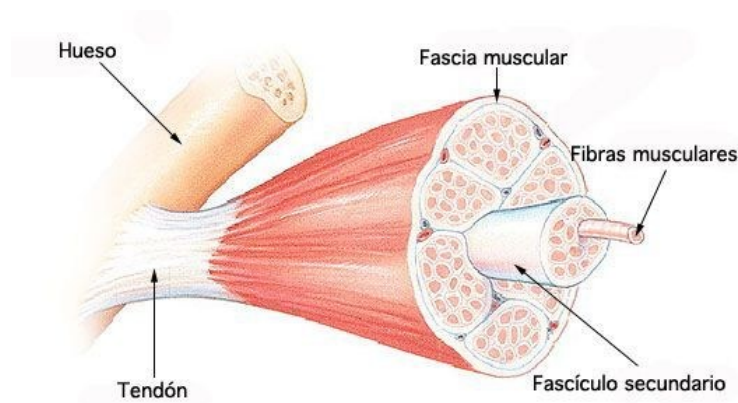


Figura 1: Composición del músculo esquelético.

Fuente: Doctordulitel (2015). Recuperado el 22 de julio, 2025.

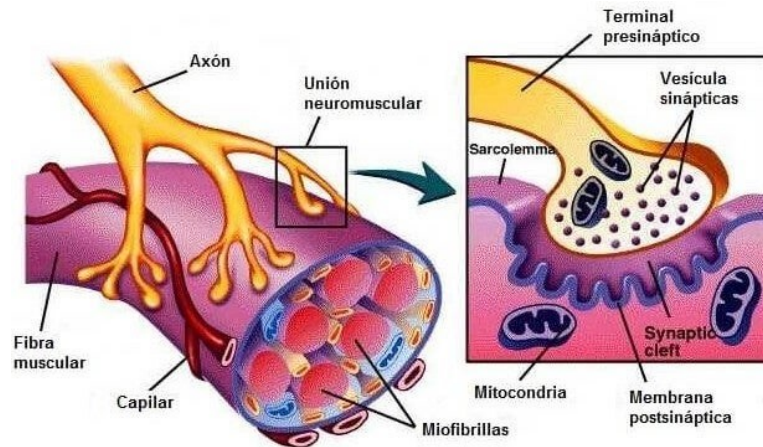


Figura 2: Unión neuromuscular.

Fuente: López (2023). Recuperado el 22 de julio, 2025.

Las neuronas motoras son células especializadas en generar y transmitir impulsos nerviosos en respuesta a un estímulo. En reposo, su membrana mantiene una diferencia de potencial eléctrico entre el interior y el exterior, con el interior más negativo debido a la distribución desigual de iones y la acción de la bomba sodio-potasio, alcanzando un valor aproximado de -70 mV.

Cuando un estímulo alcanza el umbral de excitación, cercano a -55 mV, se abren canales de sodio (Na^+) dependientes de voltaje, permitiendo la entrada rápida de estos iones y provocando la despolarización de la membrana (Potencial de Acción, 2023). Este cambio eléctrico, denominado potencial de acción nervioso, se propaga a lo largo del axón de la neurona presináptica hasta la unión neuromuscular.

Allí, la llegada del impulso abre canales de calcio (Ca^{2+}) dependientes de voltaje, lo que desencadena la liberación de acetilcolina al espacio sináptico. Este neurotransmisor se une a receptores nicotínicos en la membrana de la fibra muscular, generando un potencial de acción muscular que, si alcanza el umbral, provoca la liberación de calcio en el citoplasma e inicia la contracción de las miofibrillas que componen la fibra muscular (Visible Body, part of Cengage Learning, s. f.).

Cuando el estímulo cesa, los canales de sodio se cierran y se abren los de potasio (K^+), permitiendo la salida de estos iones al medio extracelular y favoreciendo la repolarización. En muchos casos, la salida de potasio es tan marcada que el potencial de membrana se hace más

La electromiografía (EMG) registra la suma espacial y temporal de múltiples PAUMs generados por diferentes unidades motoras durante la contracción muscular (Guzmán-Muñoz y Méndez-Rebolledo, 2018). En este registro, cada pico corresponde a un PAUM, y la señal completa se forma por la superposición de PAUMs procedentes de distintas unidades motoras, activadas con diferente sincronía e intensidad, como muestra la Figura 4.

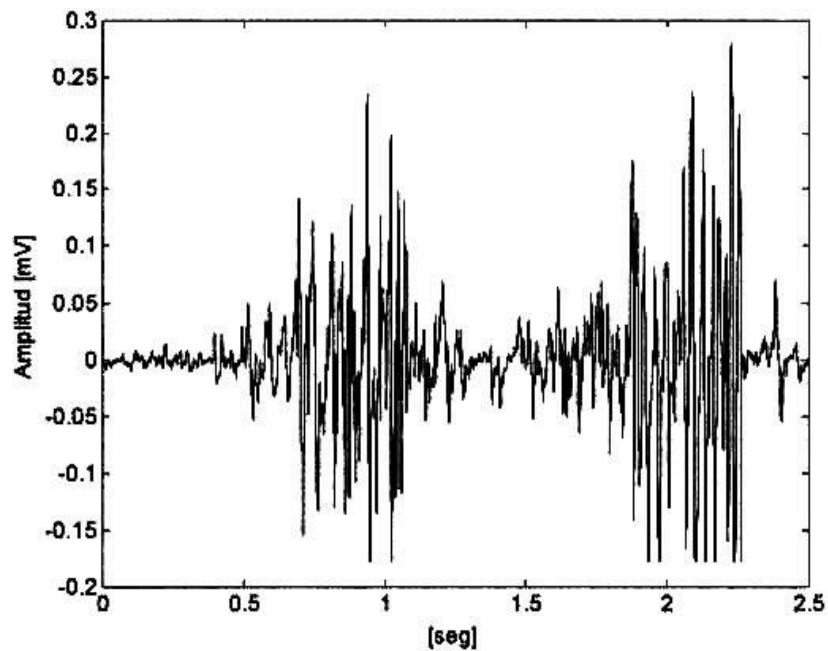


Figura 4: La electromiografía, suma de los potenciales de acción de distintas unidades motoras.

Fuente: Acevedo, R. C. et al. (2017) Recuperado el 22 de julio, 2025.

2.3. Características de la señal EMG

La señal EMG es un biopotencial de baja amplitud originado por la despolarización y repolarización de las membranas celulares de las fibras musculares. Su naturaleza es variable y estocástica, ya que resulta de la suma algebraica de múltiples potenciales de acción de unidades motoras (PAUM) que se activan de forma asincrónica.

Al no ser una señal periódica regular, su morfología varía en función del tiempo, del tipo de contracción y de la activación neuromuscular. Este carácter aleatorio, junto con su condición de señal no estacionaria (sus propiedades cambian en el tiempo), hace necesario el empleo de métodos estadísticos y de análisis espectral para su procesamiento (Del Olmo y Domingo, 2020; Reaz et al., 2006).

En condiciones fisiológicas normales, la amplitud pico a pico de la señal EMG oscila entre 0 y 10 milivoltios (-5 mV, +5 mV). Esta variabilidad depende de factores como la masa muscular, la profundidad de las fibras activadas, el tipo de electrodo utilizado (superficial o intramuscular) y la resistencia de contacto piel-electrodo. Además, existe una marcada variabilidad interindividual, determinada por diferencias anatómicas (grosor de tejido subcutáneo, longitud y orientación de las fibras, distribución de puntos motores) y fisiológicas (tipo de fibras musculares predominantes, nivel de entrenamiento, fatiga). En contracciones suaves o en músculos pequeños predominan amplitudes cercanas al rango inferior, mientras que en contracciones máximas o en músculos grandes puede alcanzarse el rango superior (Reaz et al., 2006).

En cuanto al contenido espectral, la señal EMG se distribuye principalmente entre 0 y 500 Hz, con una mayor densidad espectral de potencia en el rango 50–150 Hz, como muestra la Figura 5. Esta banda central está asociada a la frecuencia de descarga y a la velocidad de conducción de los potenciales de acción musculares.

Las componentes por debajo de 10 Hz suelen atribuirse a artefactos de movimiento o variaciones lentas en la línea base, mientras que por encima de 500 Hz predomina el ruido eléctrico y el contenido informativo es mínimo.

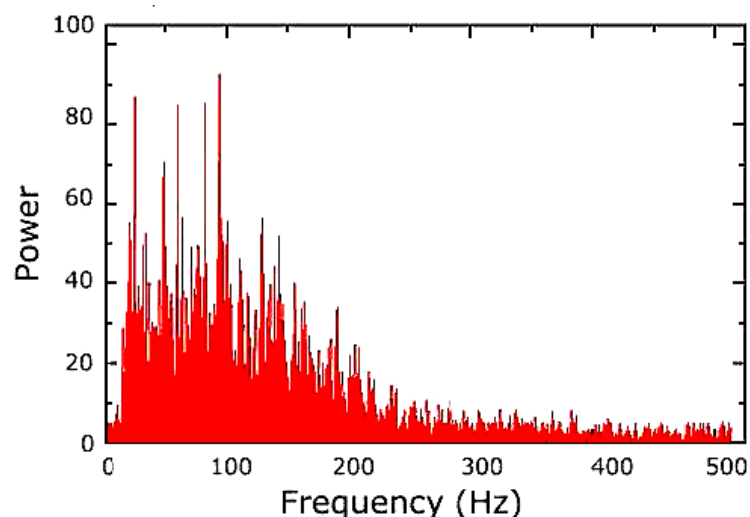


Figura 5: Espectro de frecuencias de la señal EMG.

Fuente: Shaw, L. et al. (2012). Recuperado el 23 de julio, 2025.

Según Reaz et al. (2006), la señal EMG está altamente expuesta a diferentes fuentes de ruido y distorsión, entre las que destacan cuatro:

- Ruido electrónico inherente al equipo: este tipo de artefacto no puede eliminarse, pero puede reducirse al utilizar instrumental de mayor calidad.
- Ruido ambiental: debido a la radiación electromagnética, este tipo de interferencia es prácticamente imposible de eliminar por los campos electromagnéticos constantemente ejercidos entre la superficie corporal y la del planeta, cuya amplitud es tres veces mayor a la de la señal EMG.
- Artefactos de movimiento: se generan principalmente por una preparación inadecuada de la piel o por deficiencias en el montaje del equipo. Su origen está relacionado con una elevada impedancia en la interfaz piel-electrodo y con el desplazamiento de los cables durante el registro, lo que introduce variaciones no deseadas en la señal.
- Inestabilidad intrínseca de la señal: la amplitud de la EMG varía de forma aleatoria incluso bajo condiciones de contracción estable, como consecuencia del patrón de descarga de las unidades motoras, cuya frecuencia habitual oscila entre 0 y 20 Hz. Estas componentes lentas de la señal, propias del control neuromuscular, se consideran ruido fisiológico en el análisis de EMG, ya que pueden enmascarar la actividad de interés localizada en frecuencias superiores. Para minimizar su impacto, es común aplicar filtrado pasa-alto.

3

Anatomía neuromuscular del antebrazo

El antebrazo es una región anatómica de alta complejidad funcional, en la que interactúan músculos, tendones y nervios para generar los movimientos de la muñeca, la mano y los dedos. Esta estructura permite realizar desde acciones de gran fuerza, como la prensión, hasta movimientos finos y precisos, como la pinza digital, esenciales para la manipulación de objetos y la interacción con el entorno. Su papel es fundamental en los movimientos de la vida cotidiana, tales como agarrar bolsas, abrir frascos o atarse los cordones, actividades que requieren una integración precisa de fuerza y destreza manual.

En el contexto de la detección de gestos mediante electromiografía de superficie, el conocimiento detallado de la anatomía funcional del antebrazo resulta fundamental. La identificación de los músculos responsables de cada gesto, su inervación y la localización superficial de sus vientres musculares permite seleccionar con precisión las zonas de colocación de electrodos, optimizando así la calidad de la señal registrada y reduciendo el riesgo de interferencias por la actividad de músculos adyacentes (*crosstalk*).

Para el presente trabajo se han seleccionado siete gestos, cada uno de ellos representativo de diferentes patrones de activación muscular. En la Tabla 1 se recogen los movimientos analizados durante el estudio, acompañados de su correspondiente representación gráfica para facilitar su identificación.

Movimiento	Representación del movimiento
Extensión de los dedos/apertura de la mano	 <p data-bbox="963 1234 1235 1256">Figura 6: Apertura de mano</p>
Flexión de los dedos/cierre de la mano	 <p data-bbox="978 1563 1220 1585">Figura 7: Cierre de mano</p>
Pronación del antebrazo	 <p data-bbox="935 1917 1267 1939">Figura 8: Pronación del antebrazo</p>

<p>Supinación del antebrazo</p>	 <p>Figura 9: Supinación del antebrazo</p>
<p>Posición neutra/relajada</p>	 <p>Figura 10: Posición neutra</p>
<p>Extensión de pulgar, con flexión del resto de los dedos</p>	 <p>Figura 11: Extensión de pulgar</p>
<p>Extensión y abducción del segundo y tercer dedos, con flexión de los restantes</p>	 <p>Figura 12: Extensión-abducción del segundo y tercer dedos</p>

Tabla 1: Movimientos realizados durante el estudio

Todos los movimientos parten de la posición neutra o relajada, por lo tanto, en el caso de los movimientos de pronación y supinación, el antebrazo es girado para situar la palma hacia abajo o arriba desde esa posición, respectivamente.

El antebrazo está dividido en tres compartimentos según su posición anatómica: compartimento anterior, compartimento posterior y compartimento lateral, como muestra el corte transversal de la Figura 13.

El compartimento anterior, también denominado compartimento flexor, alberga los músculos flexores y pronadores del antebrazo, mientras que el compartimento posterior o extensor, contiene los músculos extensores y supinadores (Músculos del Antebrazo, 2023). Ambos compartimentos se dividen, a su vez, en capa superficial y capa profunda.

Por su parte, el compartimento lateral, en algunas literaturas considerado como la capa lateral del compartimento posterior, contiene dos músculos extensores de la muñeca y un flexor del antebrazo (Nysora, 2022).

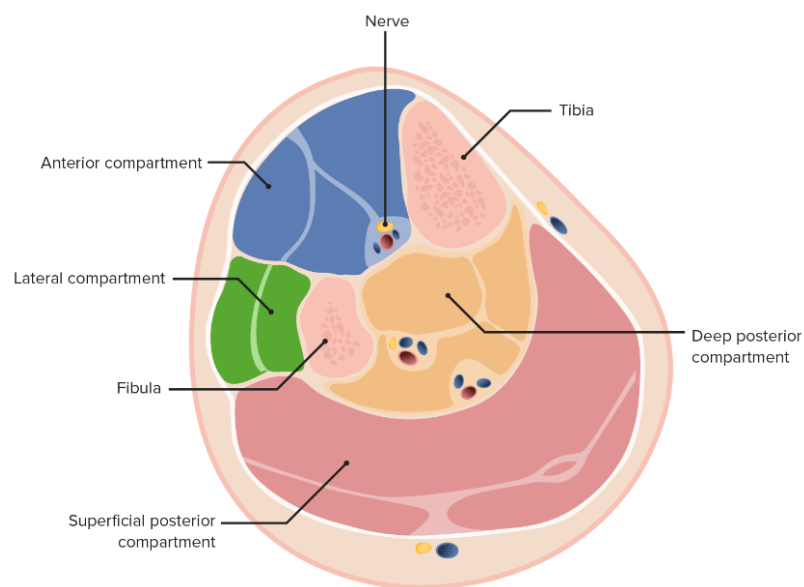


Figura 13: Compartimentos del antebrazo.

Fuente: Oiseth et al. (2025). Recuperado el 25 de julio, 2025.

Esta distribución se refleja en la Tabla 2, donde se agrupan los músculos del antebrazo que intervienen en la realización de los movimientos del estudio por compartimento y función (Músculos del Antebrazo, 2023; Movimiento Articular, s. f.).

Compartimento	Función	Músculo
Compartimento anterior	Músculos flexores y pronadores	Palmar largo
		Flexor superficial de los dedos
		Flexor profundo de los dedos
		Flexor largo del pulgar
		Flexor corto del pulgar
		Pronador cuadrado
		Pronador redondo
Compartimento posterior	Músculos extensores y supinadores	Extensor común de los dedos
		Extensor largo del pulgar
		Extensor corto del pulgar
		Extensor propio del índice
		Extensor del dedo meñique
		Supinador
Compartimento lateral	Músculos extensores de la muñeca y flexor del antebrazo	Braquiorradial
		Extensor ulnar del carpo
		Extensor radial corto del carpo
		Extensor radial largo del carpo

Tabla 2: Músculos del antebrazo agrupados por compartimento y función.

Por otra parte, las Figuras 14 y 15 representan un corte transversal y un corte frontal del antebrazo, donde pueden apreciarse la mayoría de los músculos que forman parte del estudio.

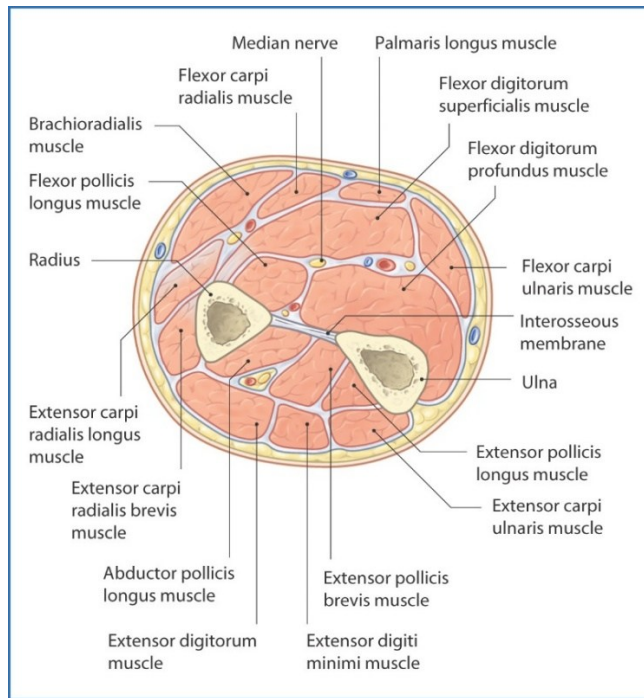


Figura 14: Músculos del antebrazo (corte transversal).

Fuente: Themes (2016). Recuperado el 30 de julio, 2025.

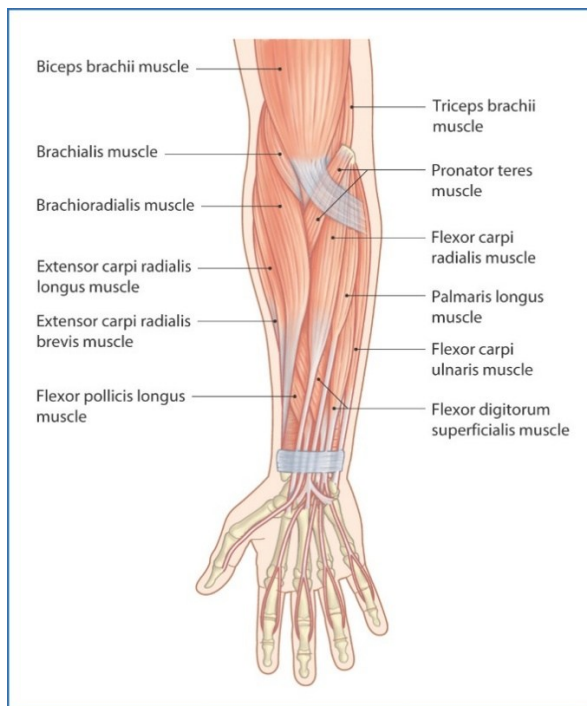


Figura 15: Músculos del antebrazo (corte frontal).

Fuente: Themes (2016). Recuperado el 30 de julio, 2025

A continuación, se describen cada uno de los músculos presentados en la Tabla 2 y en las Figuras 14 y 15, indicando su origen, inserción y movimientos que permiten llevar a cabo.

3.1 Compartimento anterior

- **Palmar largo (Palmaris longus)**: flexor superficial del compartimento anterior del antebrazo, junto al flexor radial del carpo y el flexor ulnar del carpo. Se origina en el epicóndilo medial del húmero, cruza el brazo desde el codo hasta la muñeca, su tendón atraviesa el retináculo flexor y luego se continúa con la aponeurosis palmar media. Su función principal es la de flexionar el antebrazo y la muñeca, permitiendo cerrar el puño y agarrar objetos con precisión. Además, contribuye en la estabilización de la muñeca. El 14% de la población no tiene este músculo, dado que se ha ido perdiendo evolutivamente. Su función se debía a la proporción de una fuerza extra para trepar árboles (Body Anatomy: Upper Extremity Muscles | The Hand Society, s. f.; Movimiento Articular, s. f.).
- **Flexor superficial de los dedos (Flexor digitorum superficialis)**: se encuentra en la segunda capa del compartimento anterior del antebrazo, por encima del flexor profundo de los dedos y del flexor largo del pulgar. Tiene dos cabezas, una se origina en el epicóndilo humeral y la otra en la cara anterior del radio. Se extiende por la cara anterior del antebrazo hasta los dedos. Ambas cabezas se unen y se ramifican en 4 fascículos musculares que, atravesando el túnel carpiano, darán lugar a los tendones flexores de los dedos. Estos se insertan en la falange media de cada dedo (índice, corazón, anular y meñique). Es el flexor principal de las articulaciones interfalángicas proximales (entre falange proximal y media), de los metacarpos y de la muñeca, por lo que permite cerrar el puño (Body Anatomy: Upper Extremity Muscles | The Hand Society, s. f.; Movimiento Articular, s. f.).
- **Flexor profundo de los dedos (Flexor digitorum profundus)**: flexor profundo del compartimento anterior del antebrazo, junto al flexor largo del pulgar. Se origina en el extremo distal del cúbito y la membrana interósea, se extiende desde el codo hasta los dedos, se divide en 4 fascículos musculares y sus tendones atraviesan el retináculo

flexor del túnel carpiano para insertarse en la base anterior de la falange distal de los dedos índice, corazón, anular y meñique. Flexiona estos cuatro dedos, permitiendo agarrar objetos, así como flexionar la muñeca (Body Anatomy: Upper Extremity Muscles | The Hand Society, s. f.; Movimiento Articular, s. f.).

- **Flexor largo del pulgar (Flexor pollicis longus)**: se encuentra en la capa profunda del compartimento anterior del antebrazo. Se origina en la cara anterior del radio y en la membrana interósea, atraviesa el retináculo flexor y se inserta en la base anterior de la falange distal del pulgar. Flexiona el pulgar y permite realizar el movimiento de pinza de la mano, así como la flexión de muñeca (Body Anatomy: Upper Extremity Muscles | The Hand Society, s. f.; Movimiento Articular, s. f.).
- **Flexor corto del pulgar (Flexor pollicis brevis)**: se encuentra en el compartimento anterior del antebrazo y tiene dos cabezas, una superficial y una profunda. La cabeza superficial se origina en el retináculo flexor y en el tubérculo del trapecio y la profunda se origina en los huesos grande y trapezoide. Ambas cabezas se insertan en la base de la falange proximal del pulgar. Su función principal es la de flexionar las articulaciones metacarpofalángica y carpometacarpiana, provocando la oposición y flexión del pulgar. Gracias a esta función es posible realizar actividades como el agarre de objetos o el pinzamiento (Flexor Pollicis Brevis, 2020).
- **Pronador cuadrado (Pronator quadratus)**: Se encuentra en el compartimento anterior del antebrazo. Se origina en la cara anterior del extremo distal del cúbito, se desplaza transversalmente y se inserta en el extremo distal de la cara anterior del radio. Al igual que el pronador redondo, su función principal es la pronación del antebrazo (Body Anatomy: Upper Extremity Muscles | The Hand Society, s. f.; Movimiento Articular, s. f.).
- **Pronador redondo (Pronator teres)**: Se encuentra en la capa superficial del compartimento anterior del antebrazo. Se origina en el epicóndilo medial, cruza el antebrazo y se inserta en la cara externa del cuerpo medio del radio. Gira el antebrazo para que la palma mire hacia abajo, es decir, permite el movimiento de pronación

(Body Anatomy: Upper Extremity Muscles | The Hand Society, s. f.; Movimiento Articular, s. f.).

En la imagen A de la Figura 16 pueden observarse los orígenes e inserciones de cada uno de los músculos del compartimento anterior, coloreados de naranja y azul, respectivamente. Por otro lado, las imágenes siguientes representan la disposición de los músculos por capas, comenzando por las capas superficiales a partir de la imagen B, es decir, las más cercanas a la piel y finalizando por la capa más cercana al hueso en la imagen E.

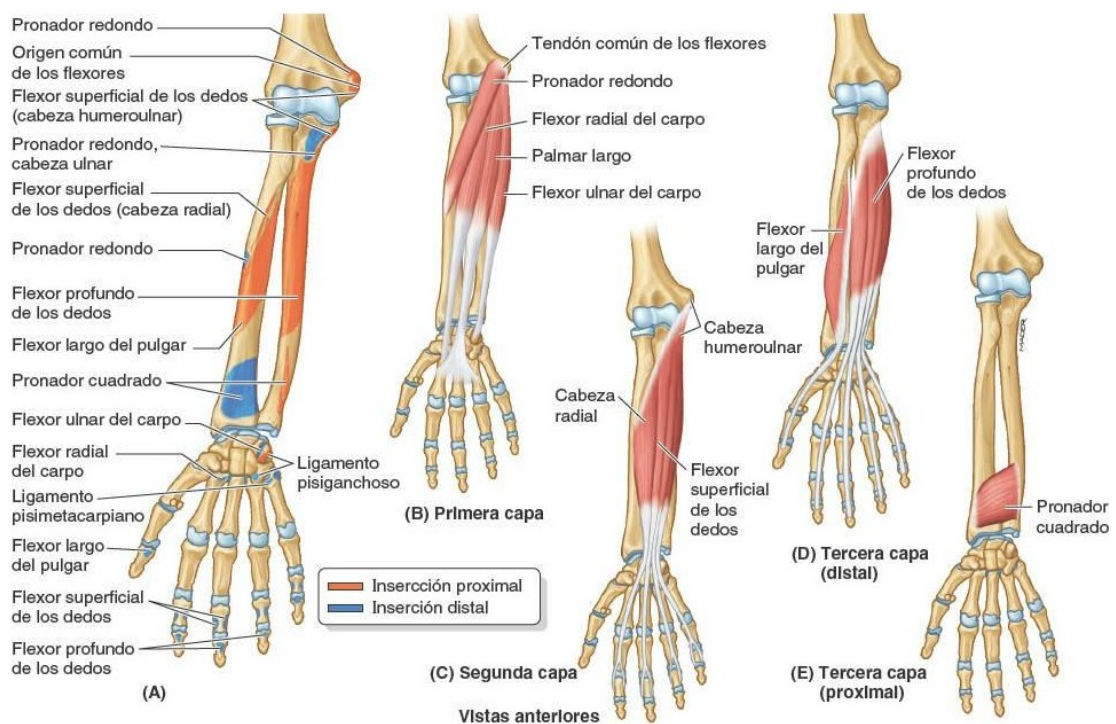


Figura 16: Músculos flexores y pronadores del antebrazo (compartimento anterior).

Fuente: Themes (2016). Recuperado el 30 de julio, 2025.

3.2 Compartimento posterior

- **Extensor común de los dedos (Extensor digitorum):** Se encuentra en la capa superficial del compartimento posterior del antebrazo. Se origina en el epicóndilo lateral del húmero, a través del tendón común de los extensores, desciende por la parte posterior del antebrazo y su vientre se ramifica en 4 tendones, que atravesarán el retináculo extensor y se insertarán en la base de la falange distal de los dedos índice,

corazón, anular y meñique. Su función consiste en la extensión de la falange proximal de los dedos, permitiendo la apertura de la mano (Body Anatomy: Upper Extremity Muscles | The Hand Society, s. f.; Movimiento Articular, s. f.).

- **Extensor largo del pulgar (Extensor pollicis longus):** Se encuentra en el compartimento posterior del antebrazo. Se origina en la diáfisis posterior del cúbito, su tendón pasa por el retináculo extensor y se inserta en la base de la falange distal del pulgar. Su función es la de extender la punta del pulgar y, junto al extensor corto, permite realizar el movimiento conocido como “thumbs up” o extensión del pulgar (Body Anatomy: Upper Extremity Muscles | The Hand Society, s. f.; (Movimiento Articular, s. f.).
- **Extensor corto del pulgar (Extensor pollicis brevis):** se encuentra en la capa profunda del compartimento posterior del antebrazo. Se origina en la cara posterior del tercio inferior del radio y en la membrana interósea, se extiende por el radio, su tendón pasa a través del retináculo extensor del túnel carpiano y se inserta en la base de la falange proximal del pulgar. Su función principal es la de extender la primera falange del pulgar, permitiendo realizar el movimiento conocido como “thumbs up” o extensión del pulgar (Body Anatomy: Upper Extremity Muscles | The Hand Society, s. f.; Movimiento Articular, s. f.).
- **Extensor propio del índice (Extensor indicis proprius):** Se encuentra en la capa profunda del compartimento posterior del antebrazo. Se origina en la parte inferior de la cara posterior del cúbito, su tendón atraviesa el retináculo extensor y se inserta en la articulación metacarpofalángica del índice. Al igual que el extensor del dedo meñique, este músculo extiende el dedo índice de forma aislada (Body Anatomy: Upper Extremity Muscles | The Hand Society, s. f.; Movimiento Articular, s. f.).
- **Extensor del dedo meñique (Extensor digiti minimi):** Se encuentra en la capa superficial del compartimento posterior del antebrazo. Se origina en el epicóndilo lateral del húmero, a través del tendón común de los extensores, su tendón atraviesa el retináculo extensor y se inserta en la base de la primera falange del meñique. Su

función consiste en la extensión del quinto dedo de forma aislada (Body Anatomy: Upper Extremity Muscles | The Hand Society, s. f.; Movimiento Articular, s. f.).

- **Extensor ulnar del carpo (Extensor carpi ulnaris)**: Se encuentra en la capa superficial del compartimento posterior del antebrazo. Se origina en el epicóndilo lateral del húmero, a través del tendón común de los extensores y pasa por encima del cúbito. Su tendón atraviesa el retináculo extensor y se inserta en la cara interna de la base del quinto metacarpiano. Su función, junto con los extensores radiales del carpo, es la de enderezar y estabilizar la muñeca al cerrar el puño con fuerza, compensando la fuerza de los músculos flexores para evitar su colapso (Body Anatomy: Upper Extremity Muscles | The Hand Society, s. f.; *Movimiento Articular*, s. f.).
- **Supinador (Supinator)**: Se encuentra en la capa profunda del compartimento posterior del antebrazo. Se origina en el epicóndilo lateral del húmero y se inserta en la cara externa del cuerpo medio del radio. Gira el antebrazo para que la palma mire hacia arriba, es decir, permite el movimiento de supinación del antebrazo (Body Anatomy: Upper Extremity Muscles | The Hand Society, s. f.; *Movimiento Articular*, s. f.).

3.3 Compartimento lateral

- **Braquiorradial (Brachioradialis)**: Se encuentra en el compartimento lateral del antebrazo. Se origina en la cresta supracondílea lateral del húmero, desciende sobre la cara externa del radio y se inserta en la superficie lateral del extremo distal del radio. Supina y prona el antebrazo desde su posición máxima en los primeros grados de giro, después actúan el supinador y el pronador (Body Anatomy: Upper Extremity Muscles | The Hand Society, s. f.; *Movimiento Articular*, s. f.).
- **Extensor radial largo del carpo (Extensor carpi radialis longus)**: Se encuentra en el compartimento lateral del antebrazo. Se origina en el epicóndilo lateral del húmero, a través del tendón común de los extensores, su tendón atraviesa el retináculo extensor y se inserta en la cara dorsal de la base del tercer metacarpiano. Su función consiste en compensar la fuerza de los músculos flexores al cerrar el puño, junto con el extensor

radial corto del carpo (Body Anatomy: Upper Extremity Muscles | The Hand Society, s. f.; Movimiento Articular, s. f.).

- **Extensor radial corto del carpo (Extensor carpi radialis brevis):** Se encuentra en el compartimento lateral del antebrazo. Se origina en el epicóndilo lateral del húmero, a través del tendón común de los extensores, su tendón atraviesa el retináculo extensor y se inserta en la cara dorsal de la base del segundo metacarpiano. Su función consiste en compensar la fuerza de los músculos flexores al cerrar el puño, junto con el extensor ulnar del carpo y extensor radial largo del carpo (Body Anatomy: Upper Extremity Muscles | The Hand Society, s. f.; Movimiento Articular, s. f.).

En la imagen A de la Figura 17 quedan representados los orígenes e inserciones de los músculos extensores y supinadores, pertenecientes a los compartimentos posterior y lateral del antebrazo, agrupados con el nombre de compartimento extensor. Además, en las imágenes siguientes pueden apreciarse la disposición de los músculos por capas, comenzando por la capa superficial en la imagen B y finalizando con la capa profunda en la imagen C.

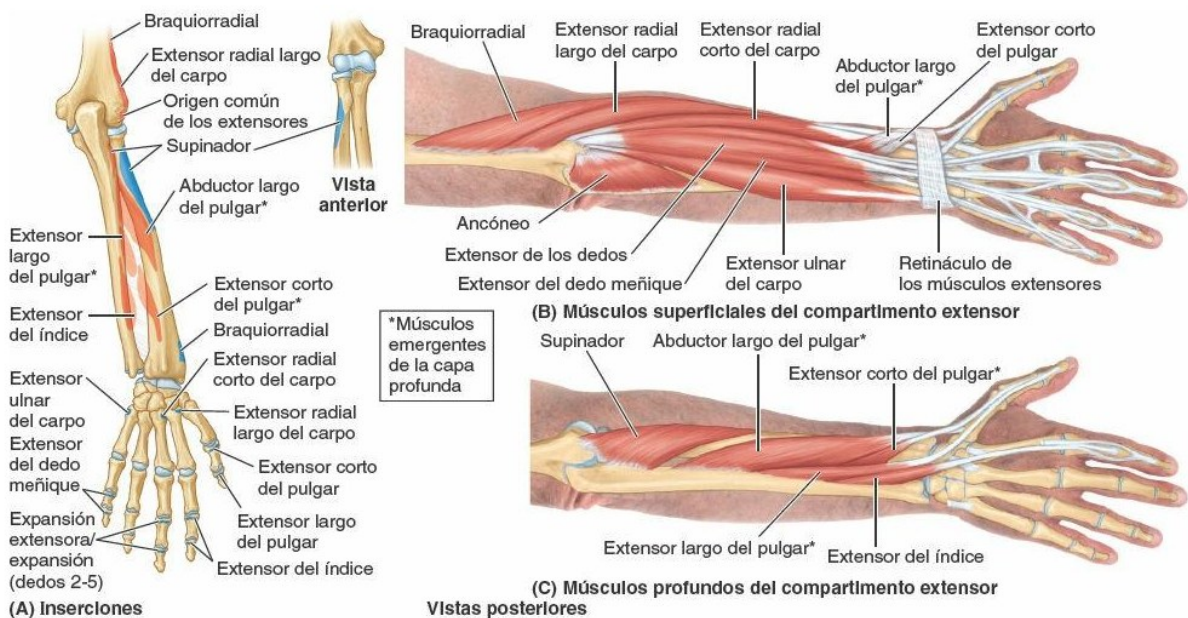


Figura 17: Músculos extensores, braquiorradial y supinador del antebrazo (compartimento posterior y lateral).

Fuente: 08. Antebrazo | Enfermería (2024). Recuperado el 30 de julio, 2025.

3.4 Inervación de los músculos del antebrazo

Para que los músculos del antebrazo puedan contraerse y generar los movimientos correspondientes, requieren ser activados por un potencial de acción, tal como se explicó en el apartado 2.2. Esta activación depende de la inervación nerviosa, que varía en función del compartimento anatómico en el que se localice cada músculo.

Los músculos flexores y pronadores, situados en el compartimento anterior, reciben inervación principalmente del nervio mediano, con la excepción de la mitad medial del flexor profundo de los dedos, que está inervada por el nervio ulnar (Músculos del Antebrazo, 2023).

Por su parte, los músculos extensores y supinadores, pertenecientes al compartimento posterior, son inervados por el nervio interóseo posterior, rama terminal del nervio radial. Una excepción relevante es el músculo braquiorradial, que, aunque funcionalmente actúa como flexor, se considera anatómicamente dentro del compartimento posterior; en consecuencia, su inervación difiere de la de los flexores típicos y depende directamente del nervio radial (Músculos del Antebrazo, 2023).

La Tabla 3 presenta un resumen de la inervación de los músculos del antebrazo según su función.

Grupo de músculos según su función	Inervación	Músculo
Músculos flexores y pronadores	Nervio mediano	Pronador redondo
		Palmar largo
		Flexor superficial de los dedos
		Flexor profundo de los dedos
		Flexor largo del pulgar
		Flexor corto del pulgar
		Pronador cuadrado
	Nervio ulnar	Mitad medial del flexor profundo de los dedos
Músculos extensores y supinadores	Nervio interóseo posterior	Extensor común de los dedos
		Extensor ulnar del carpo
		Extensor del dedo meñique
		Supinador
		Extensor largo del pulgar
		Extensor corto del pulgar
		Extensor propio del índice
	Nervio radial	Braquiorradial
		Extensor radial largo del carpo
		Extensor radial corto del carpo

Tabla 3: Inervación de los músculos del antebrazo

4

Adquisición de señales

EMG

La adquisición de señales electromiográficas requiere un sistema que permita registrar de manera fiable la actividad eléctrica generada por los músculos durante la contracción. Esto resulta especialmente importante porque, como se comentó en el apartado 2.3, la señal EMG es altamente variable y puede verse afectada por un elevado nivel de ruido si no se realiza una adecuada preparación previa del registro.

Para este trabajo se utilizará una placa de circuito impreso desarrollada en el Trabajo de Fin de Grado "Diseño de un módulo de acondicionamiento para EEG", realizado por Pablo Camacho Sánchez, graduado en Ingeniería de la Salud (Camacho, 2025). El dispositivo cuenta con ocho canales de registro y un canal de referencia, lo que posibilita la adquisición simultánea de múltiples señales musculares, como muestra la Figura 18. Esta configuración resulta adecuada para el estudio de la actividad EMG en diferentes grupos musculares, facilitando además una clasificación más precisa de los gestos.

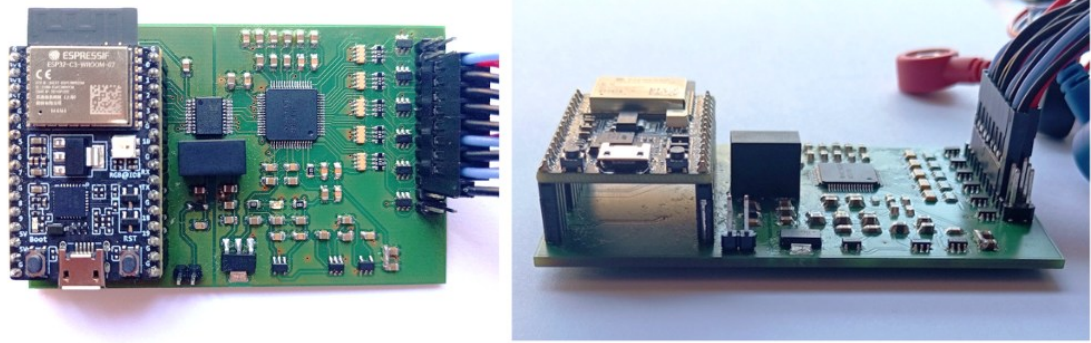


Figura 20: Placa de circuito impreso del sistema de adquisición

La transmisión de los datos se efectúa mediante protocolo SPI (Serial Peripheral Interface), caracterizado por establecer una comunicación síncrona y full dúplex entre un dispositivo maestro y uno o varios esclavos, posibilitando así un intercambio continuo de información con una alta tasa de transferencia de datos (Learn About The Serial Peripheral Interface (SPI) · VectorNav, s. f.).

Tanto la frecuencia de muestreo como la ganancia del convertor son parámetros programables. En particular, se dispone de frecuencias de muestreo de 125, 250, 500, 1000, 2000, 4000 y 8000 Hz, y de ganancias de 1, 2, 3, 4, 6, 8 y 12.

Tal como se expuso en el apartado 2.3, el espectro de la señal EMG se encuentra en el rango de 0 a 500 Hz, concentrándose la mayor parte de su energía en el intervalo comprendido entre 50 y 150 Hz. De acuerdo con Muceli & Merletti (2024) y Adquirir una Señal Analógica: Ancho de Banda, Teorema de Muestreo de Nyquist y Aliasing (2006), y conforme al teorema del muestreo de Nyquist, la frecuencia de muestreo debe ser, como mínimo, el doble de la frecuencia máxima presente en la señal muestreada. En caso contrario, se produciría el fenómeno de *aliasing*, caracterizado por la aparición de componentes de baja frecuencia que distorsionan la reconstrucción de la señal (Adquirir una Señal Analógica: Ancho de Banda, Teorema de Muestreo de Nyquist y Aliasing, 2006).

En consecuencia, la frecuencia de muestreo seleccionada para la adquisición de los datos debe ser al menos 1 kHz. No obstante, se optó por una frecuencia de 2 kHz, ya que se recomienda emplear una tasa de muestreo entre 3 y 4 veces superior a la frecuencia máxima de la señal con el fin de garantizar una reconstrucción de alta calidad. Asimismo, esto permite

reducir los requerimientos del filtrado posterior, evitando el solapamiento espectral de las señales y facilitando su separación.

En cuanto a la cuantificación, el conversor de 16 bits permite la división y representación de la amplitud de la señal analógica en 2^{16} niveles de cuantificación. El rango de entrada del ADC es bipolar y se define como $\pm 2.4V/GAIN$, tal y como se indica en el datasheet del conversor (ADS1198 Data Sheet, Product Information And Support | TI.com, s. f.), donde la ganancia es programable y atiende a los valores anteriormente expuestos.

Para ajustar el rango de amplitud de la señal EMG de la manera más adecuada posible al rango de entrada del conversor, se seleccionó la ganancia de 12. De esta forma, el rango efectivo de entrada queda reducido a $\pm 0.2 V$, maximizando así la resolución.

Por otro lado, el tamaño del escalón de cuantificación, definido como la amplitud mínima analógica que puede representar un incremento unitario en el código digital (Universitat Politècnica de València - UPV, 2015), se obtiene a partir de la relación de la Ecuación 1:

$$Q = \frac{(2 \cdot A)}{2^n} \quad (1)$$

Donde A es la amplitud física del ADC ($A=2.4 V$), n es el número de bits del conversor ($n=16$) y Q representa el valor del escalón de cuantificación. Sustituyendo en la expresión se obtiene la Ecuación 2:

$$Q = \frac{2 \cdot 2.4 V}{2^{16}} = \frac{4.8 V}{65536} = 73.2 \mu V \quad (2)$$

Teniendo en cuenta la ganancia seleccionada ($GAIN=12$), el tamaño del escalón de cuantificación queda reducido a la Ecuación 3:

$$Q = \frac{73.2 \mu V}{12} \approx 6.1 \mu V \quad (3)$$

Esto implica que cada incremento en el valor digital registrado por el ADC corresponde aproximadamente a $6.1 \mu V$ de variación de la señal analógica. Para ejemplificar, en un registro en el que la señal alcanzó un valor máximo de 1169 cuentas y un mínimo de 845 cuentas, la

diferencia de 324 unidades equivale a una amplitud física de 1.976 mV, como muestra la Ecuación 4:

$$1169 - 845 = 324 \cdot 6.1 \mu\text{V} = 1.976 \text{ mV} \quad (4)$$

De esta forma, la cuantificación digital permite vincular directamente las cuentas generadas por el ADC con las amplitudes físicas de la señal EMG, garantizando una representación precisa y adecuada a la dinámica de la señal registrada.

Para cerrar esta etapa de caracterización del sistema de adquisición, después de seleccionar la frecuencia de muestreo, ajustar la ganancia del conversor y definir el escalón de cuantificación, se procedió a adaptar la velocidad de transmisión de datos. Dado que la frecuencia de muestreo empleada (2000 Hz) es relativamente alta para garantizar una adecuada reconstrucción de la señal EMG, resultó necesario incrementar la frecuencia de reloj del bus SPI, que en la librería utilizada adopta por defecto un valor de 400 kHz. Con el fin de transmitir todas las muestras adquiridas de forma continua y sin pérdidas de información, dicho valor se configuró finalmente en 5 MHz. De forma complementaria, la velocidad de transmisión de los datos por el puerto serie se fijó en un baudrate de 2 millones, garantizando así un envío fluido y sin interrupciones durante la adquisición.

4.2 Adquisición de datos

Previo al registro de la señal EMG, es necesaria la colocación estratégica de los electrodos de manera que el ruido captado sea mínimo. Para ello y, según las recomendaciones del proyecto SENIAM, la piel deberá estar previamente rasurada y limpia con alcohol para aumentar la adherencia de los electrodos y reducir la impedancia de la piel (Welcome To SENIAM, s. f.). Asimismo, la distancia centro a centro entre los electrodos de un mismo canal debe ser de al menos 20 mm. Esto mejorará la calidad de la señal y disminuirá los efectos del *crosstalk*.

De este modo, los electrodos se colocarán formando dos circunferencias de 8 electrodos cada una y separadas 20 mm centro a centro entre sí alrededor del antebrazo, de

manera que una de las circunferencias represente el polo positivo y otra el negativo de los canales. La posición central entre ambas circunferencias se situará a una distancia de $1/3$ de la longitud del antebrazo, medida desde el codo. Finalmente, el electrodo de referencia se situará sobre el dorso de la mano contraria a la que realizará el movimiento. Como se observa en la Figura 21, la circunferencia distal al codo está formada por los polos positivos de cada canal, mientras que la proximal conforma los polos negativos. Adicionalmente, los electrodos se colocarán en sentido contrario a las agujas del reloj, como puede observarse en la misma Figura 21.

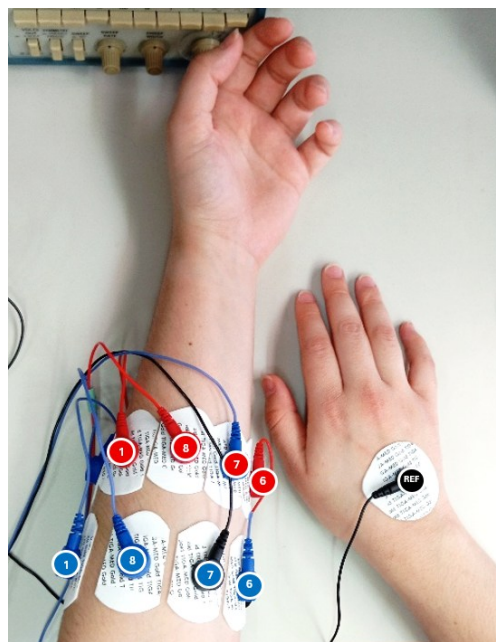


Figura 21: Colocación de los electrodos

En cuanto a la adquisición de datos, participaron en el estudio 4 usuarios de distinta edad y sexo, quienes realizaron 10 repeticiones de cada movimiento descrito en el apartado 3, obteniéndose 40 muestras por movimiento y un total de 280 muestras. Cada repetición consistió en 2 segundos de ejecución del movimiento, precedidos y sucedidos por 2 segundos de reposo. Siguiendo esta metodología, se inició la toma de datos desde la terminal de Ubuntu mediante el comando *tio*. Finalmente, se representaron gráficamente los resultados obtenidos para cada canal y movimiento en MATLAB, dando lugar a gráficas como la que se muestra en la Figura 22, que ilustra las señales en crudo expresadas en cuentas digitales, obtenidas en los ocho canales para una repetición de apertura de mano.

Aplicando la cuantificación desarrollada en el apartado 4.1 y multiplicando la señal anterior por el escalón de cuantificación ($6.1 \mu\text{V}$), se obtiene una gráfica como la que muestra la Figura 23, que mantiene la misma morfología que la de la Figura 22, pero expresada en términos de tensión. En este sentido, la señal EMG obtenida durante la apertura de mano alcanza un rango de amplitud máximo de aproximadamente 1 mV pico a pico.

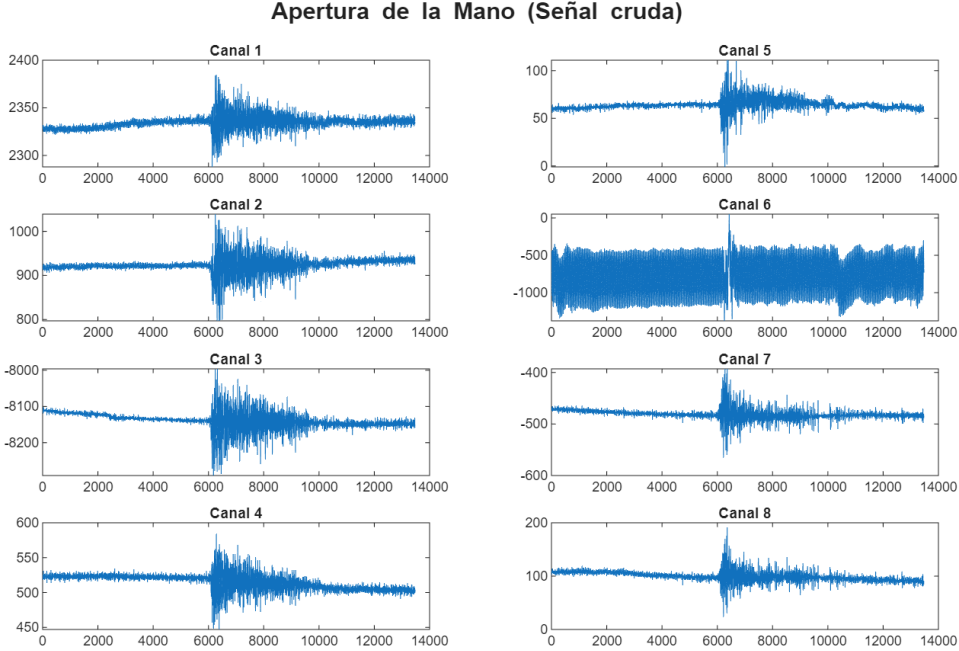


Figura 22: Apertura de la mano (Señal cruda)

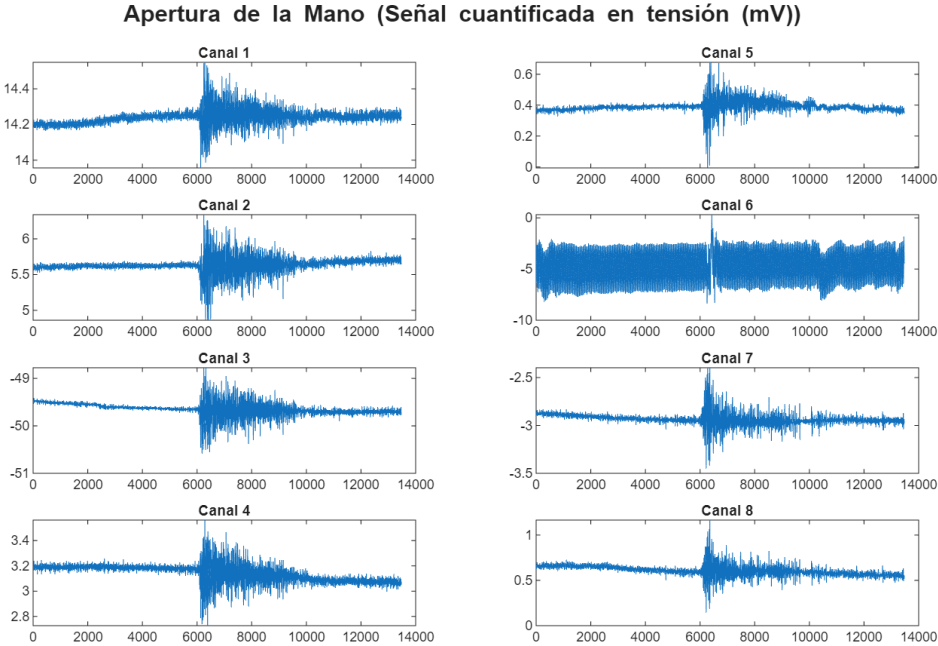


Figura 23: Apertura de la mano (Señal cuantificada en tensión)

Observando la Figura 23, se puede apreciar que la señal no se encuentra centrada en cero, sino que presenta un desplazamiento respecto al eje cero. Este hecho indica la presencia de un componente de continua (offset) que deberá corregirse mediante un proceso de centrado previo al análisis de la señal EMG, junto con un filtrado para eliminar el posible ruido presente.

5

Procesamiento de la señal EMG

Con la finalidad de garantizar la calidad y la fiabilidad de las señales electromiográficas, resulta necesario aplicar un procesamiento previo a su análisis. Este acondicionamiento permite eliminar componentes que distorsionan la señal, tales como artefactos por movimiento, ruido eléctrico o desplazamientos de base que podrían interferir en el análisis posterior. Para ello se emplea MATLAB, un entorno adecuado para implementar las distintas etapas de procesamiento y comprobar visualmente los resultados obtenidos.

En primer lugar, se lleva a cabo un centrado en cero de la señal para eliminar el ruido de corriente continua, que provoca un desplazamiento de la línea base como el que puede observarse en las gráficas de la Figura 23. Seguidamente, se aplican distintos procedimientos de filtrado, entre los que destacan el filtro Notch, usado para atenuar la interferencia de la red eléctrica, ya que produce una componente significativa a la frecuencia de 50 Hz en Europa.

Asimismo, se aplica un filtro paso banda, que permite delimitar el rango de frecuencias de interés de la señal y eliminar el ruido de alta y baja frecuencia.

Así, el procesamiento constituye una etapa imprescindible en el acondicionamiento de las señales para asegurar unas condiciones óptimas en su posterior análisis.

5.1 Centrado en cero

El ruido de corriente continua o ruido DC es un artefacto que se produce en gran medida por imperfecciones de los circuitos, errores de calibración o acoplamientos inadecuados. Este se manifiesta como la presencia de un nivel de tensión constante no deseado superpuesto a la señal principal, provocando el desplazamiento de la señal y distorsionando su procesamiento posterior (Miyara & Lahoz, 2003).

Para eliminarlo, se ha procedido a restar a cada uno de los canales de las señales su media por canal, de manera que la señal quede centrada en cero. De esta forma, el resultado de este procesamiento inicial puede observarse en las Figuras 24-30, que muestran la morfología de la señal EMG en cada movimiento diferente realizado.

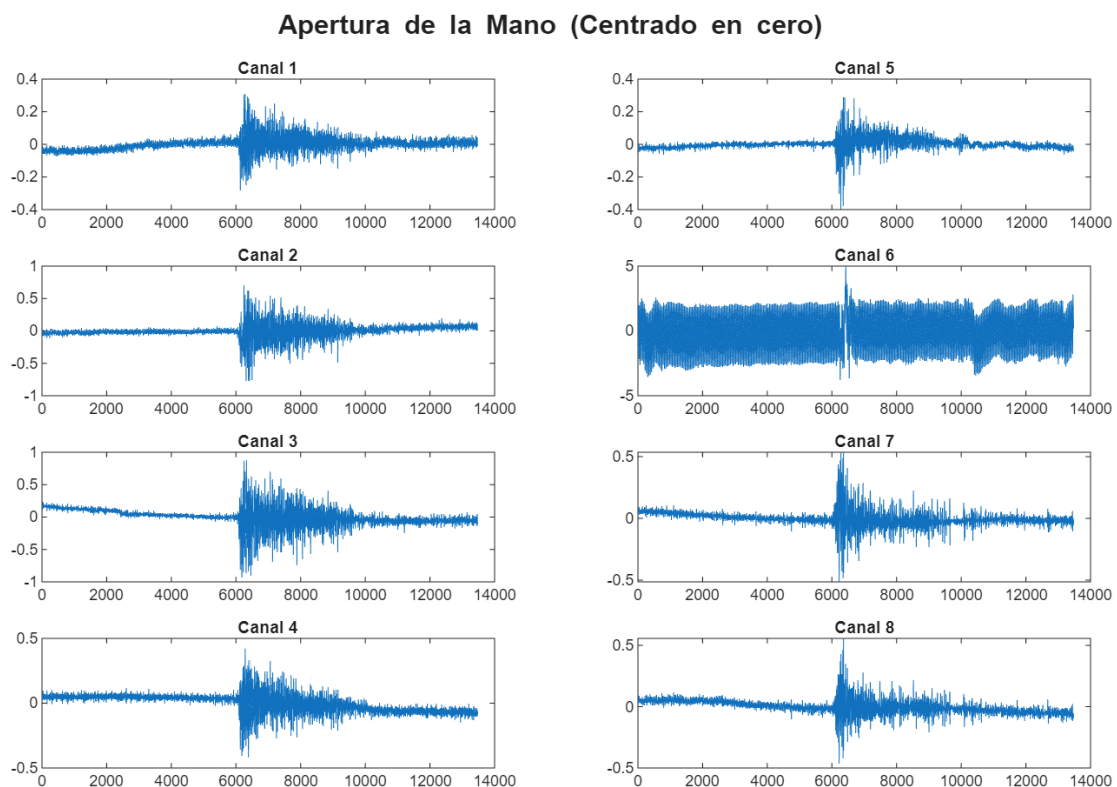


Figura 24: Apertura de la mano (Centrado en cero)

Cierre de la Mano (Centrado en cero)

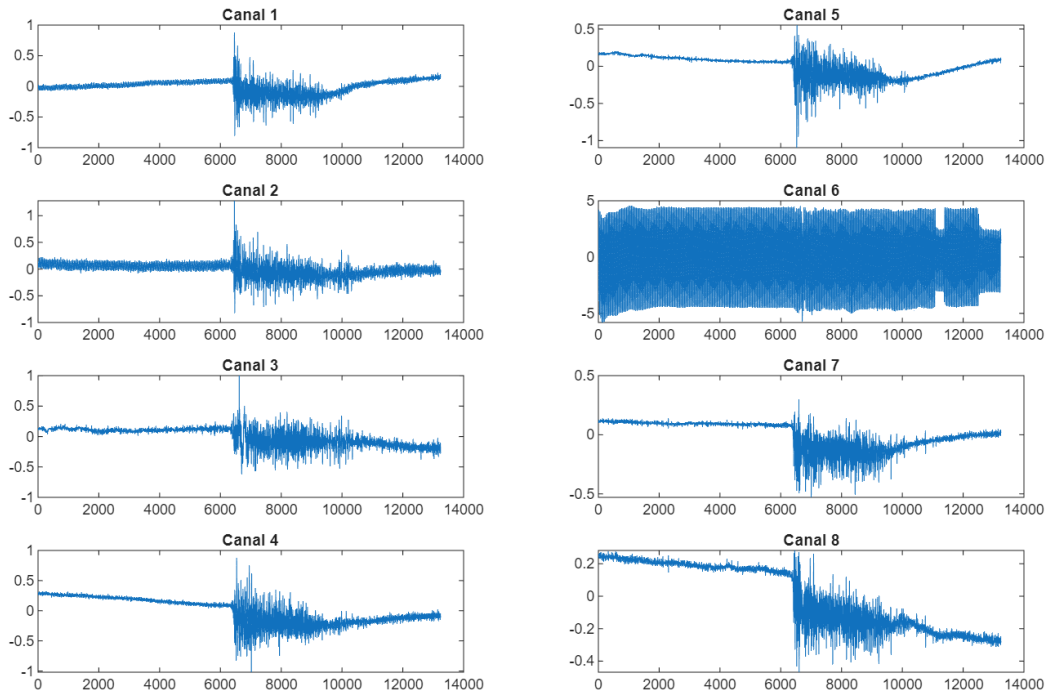


Figura 25: Cierre de la mano (Centrado en cero)

Pronación (Centrado en cero)

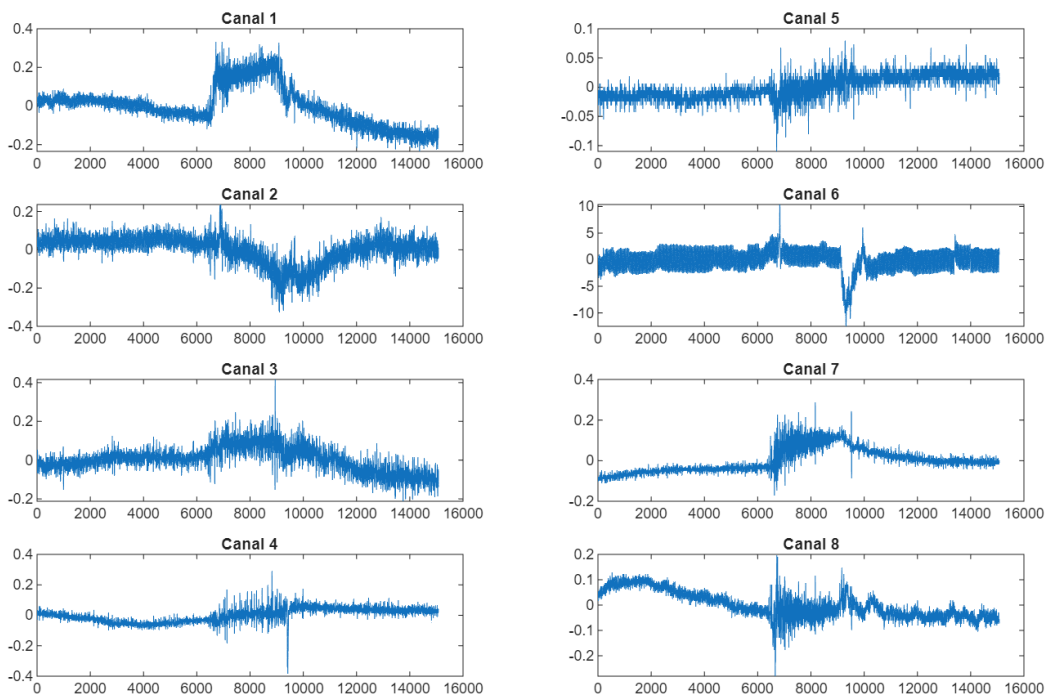


Figura 26: Pronación (Centrado en cero)

Supinación (Centrado en cero)

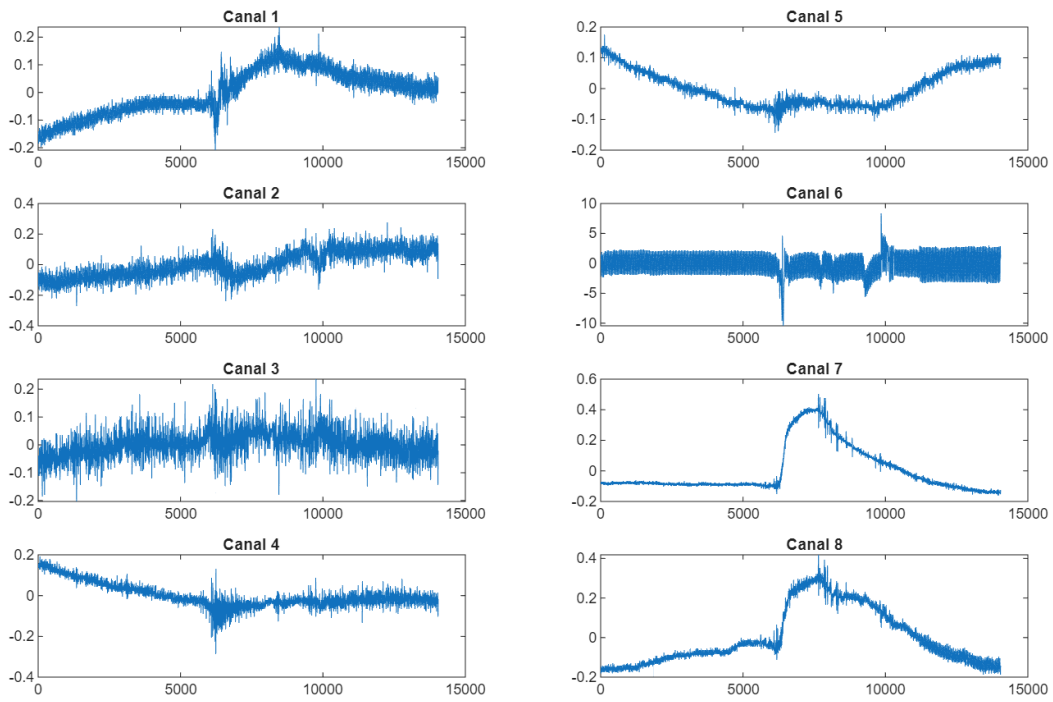


Figura 27: Supinación (Centrado en cero)

Posición neutra (Centrado en cero)

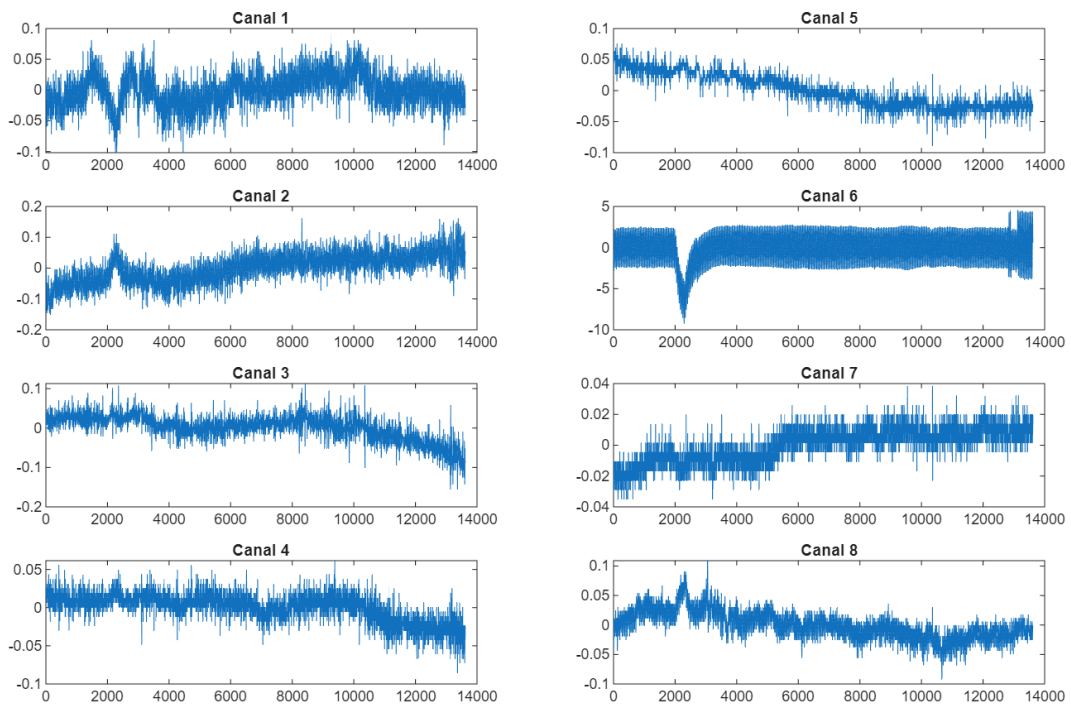


Figura 28: Posición neutra (Centrado en cero)

Extensión de pulgar (Centrado en cero)

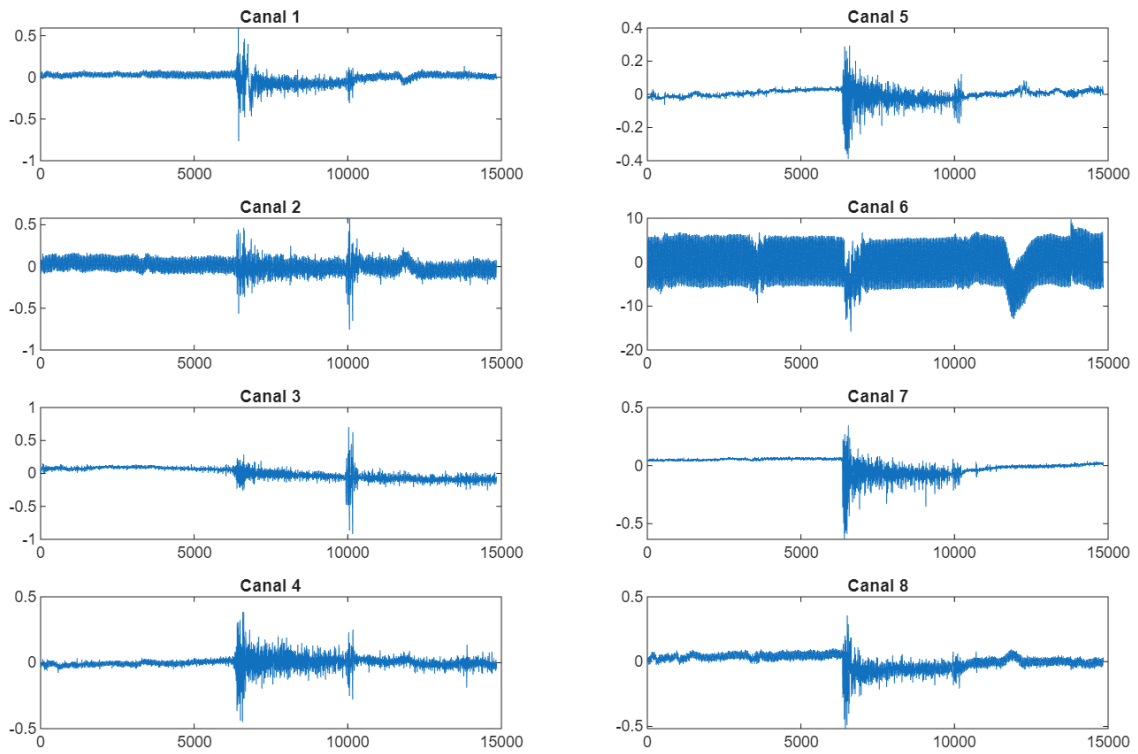


Figura 29: Extensión de pulgar (Centrado en cero)

Extensión segundo y tercer dedos (Centrado en cero)

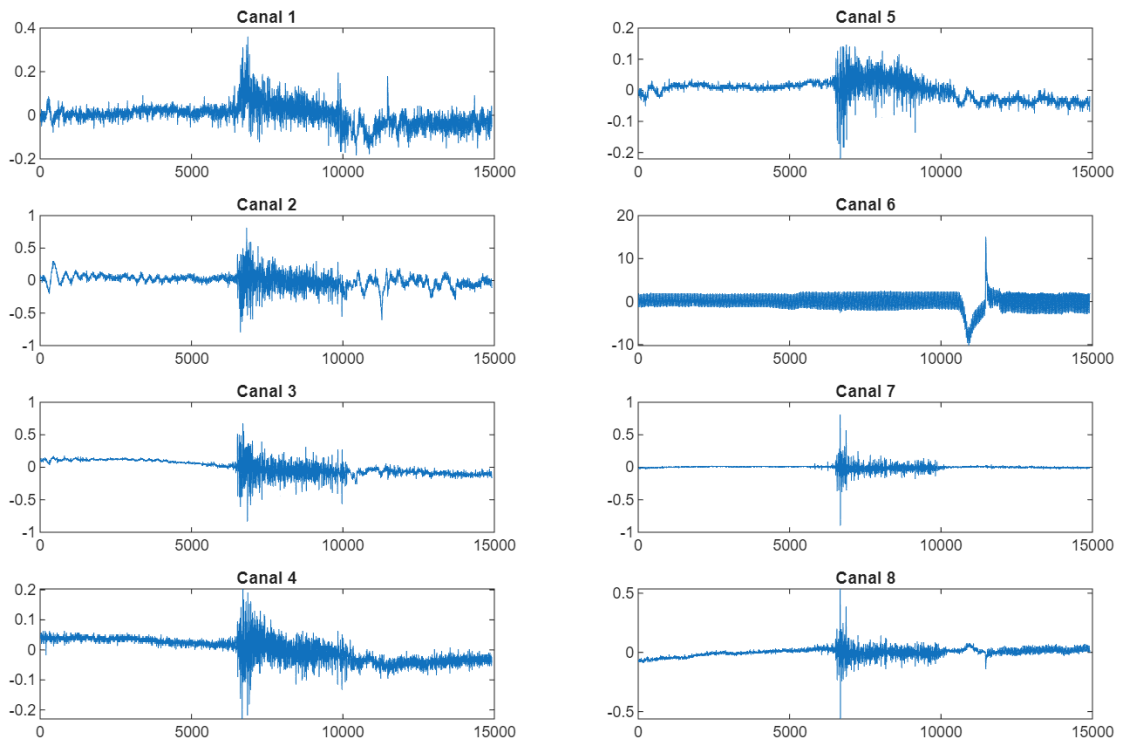


Figura 30: Extensión segundo y tercer dedos (Centrado en cero)

5.2 Filtro Notch

Una vez obtenidas las señales libres de componente continua, se procederá al diseño del filtro más adecuado para acotar el rango de frecuencias de interés, suprimiendo el posible ruido existente. En primer lugar, se deberá determinar el rango de frecuencias de las señales, que, aunque en el caso de EMG suele encontrarse entre 0 y 500 Hz, conviene precisar mejor para poder diseñar un filtro que elimine la mayor cantidad de ruido y así conseguir un procesamiento óptimo.

Para lograr este fin, se aplicará la Transformada Rápida de Fourier (FFT), la cual permite obtener la representación de una señal en el dominio de la frecuencia a partir de su expresión en el dominio del tiempo. Esta herramienta facilita la descomposición de cualquier señal compleja en una suma de ondas sinusoidales de diferentes frecuencias, mostrando cuánta energía contiene en cada una de ellas (Transformación de Fourier: Qué Es. Diccionario CUN, s. f.). De este modo, es posible identificar con claridad las frecuencias principales presentes en las señales.

Aplicando la FFT a cada uno de los canales de las señales, se obtuvieron gráficas similares a las mostradas en la Figura 31, correspondientes al canal 1 durante los movimientos de apertura de mano y en posición neutra o relajada. En el caso de la apertura de mano, puede observarse que la energía de la señal se distribuye en el rango de 0 a 500 Hz, concentrándose en su mayor parte en torno a 50 Hz. Por otro lado, en la condición de reposo también se aprecia una componente dominante en 50 Hz, lo cual resulta anómalo considerando que no debería existir actividad muscular en ese estado.

La elevada magnitud en esta frecuencia, tanto en reposo como en contracción, constituye un comportamiento atípico respecto a la actividad muscular normal registrada en una señal EMG y se atribuye a la presencia de ruido eléctrico procedente de la corriente alterna de la red eléctrica en Europa, la cual suele transmitirse a través de las tomas de corriente y ser captada por los electrodos durante la adquisición de la señal.

Las señales EMG suelen presentar una amplitud menor que la del ruido de 50 Hz, lo que provoca que este pueda enmascararlas y conducir a interpretaciones erróneas de los

resultados. Por este motivo, resulta necesario diseñar un filtro Notch o de rechazo de banda centrado en 50 Hz, con el fin de eliminar el efecto de este ruido sobre las señales.

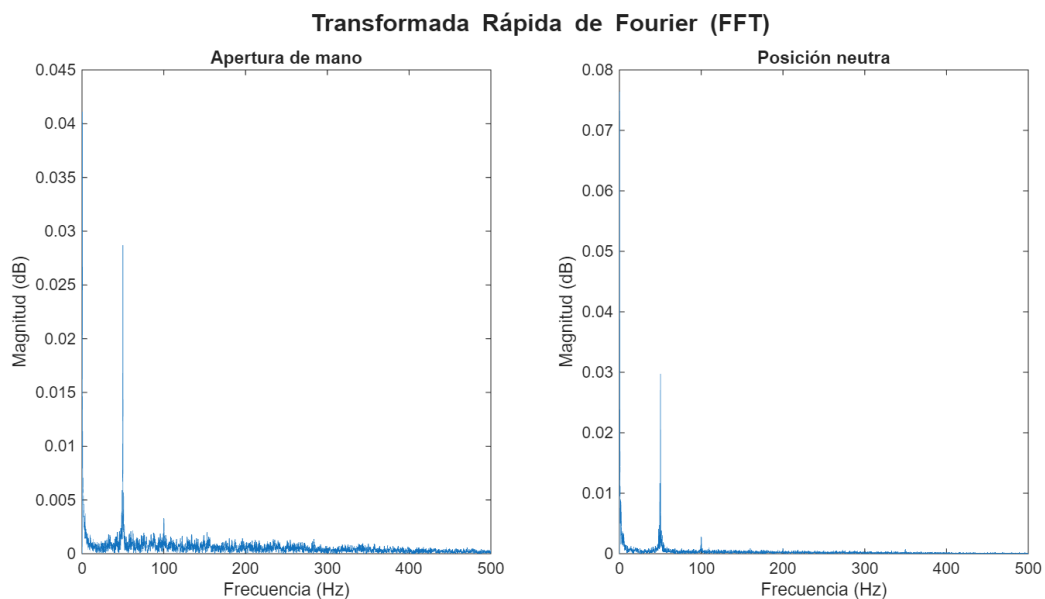


Figura 31: Comparación de la FFT en apertura y reposo

Para la eliminación del artefacto se optó por diseñar un filtro notch de tipo IIR (*Infinite Impulse Response*). Este tipo de filtros permite la atenuación de frecuencias de forma aproximada al comportamiento de un filtro ideal, consiguiendo un rechazo muy selectivo con un orden bajo, destacando así su eficiencia computacional frente a un filtro FIR (*Finite Impulse Response*) (Sörnmo & Laguna, 2005).

Por otro lado, el retardo en la salida de un filtro IIR es prácticamente inexistente, a diferencia de los filtros FIR. Asimismo, los filtros IIR ofrecen una buena resolución en bajas frecuencias, lo que permite un diseño compacto capaz de eliminar de forma selectiva la interferencia de 50 Hz sin afectar de manera significativa al resto del espectro de la señal EMG (Oppenheim et al., 1999).

Para el diseño del filtro Notch implementado en este trabajo, se utilizó la función *iirnotch* de MATLAB (*iirnotch* - (To Be Removed) Second-order IIR Notch Filter - MATLAB, s. f.), cuyos parámetros principales son la frecuencia central normalizada (w_0) y el ancho de banda normalizado (b_w).

La frecuencia central normalizada representa la frecuencia que el filtro Notch atenúa, expresada en relación con la frecuencia de Nyquist. Se define como la frecuencia a atenuar por el filtro, f_0 , dividido entre la mitad de la frecuencia de muestreo, $f_s/2$, tal como se muestra en la Ecuación (5):

$$w_0 = \frac{f_0}{(f_s/2)} \quad (5)$$

En este caso, $f_0 = 50 \text{ Hz}$ y $f_s = 2000 \text{ Hz}$, por tanto, w_0 toma un valor de 0.05.

Por su parte, el ancho de banda normalizado determina la anchura de la muesca alrededor de w_0 , es decir, el rango de frecuencias que serán atenuadas en torno a 50 Hz.

Este parámetro está relacionado con el factor de calidad (Q) del filtro, que controla la selectividad de la muesca: a valores más altos de Q , la anchura de la muesca disminuye y, en consecuencia, el filtro será más selectivo. La relación entre ambos parámetros está determinada por w_0 y se expresa en la Ecuación (6):

$$b_w = \frac{w_0}{Q} \quad (6)$$

En este caso, se adopta un valor de $Q=30$, ya que la banda a atenuar debe centrarse en 50 Hz sin extenderse demasiado. De lo contrario, podría eliminarse información relevante de la señal. Por tanto, b_w toma un valor de 0.0016.

Con estos valores, la función *iirnotch* de MATLAB calcula los coeficientes del filtro digital IIR de segundo orden. Dichos coeficientes son los que finalmente se utilizan para implementar el filtro y aplicar la atenuación de la componente en 50 Hz sobre la señal. La Figura 32 ilustra la respuesta en frecuencia del filtro, en la que se aprecia claramente la muesca centrada en la frecuencia objetivo.

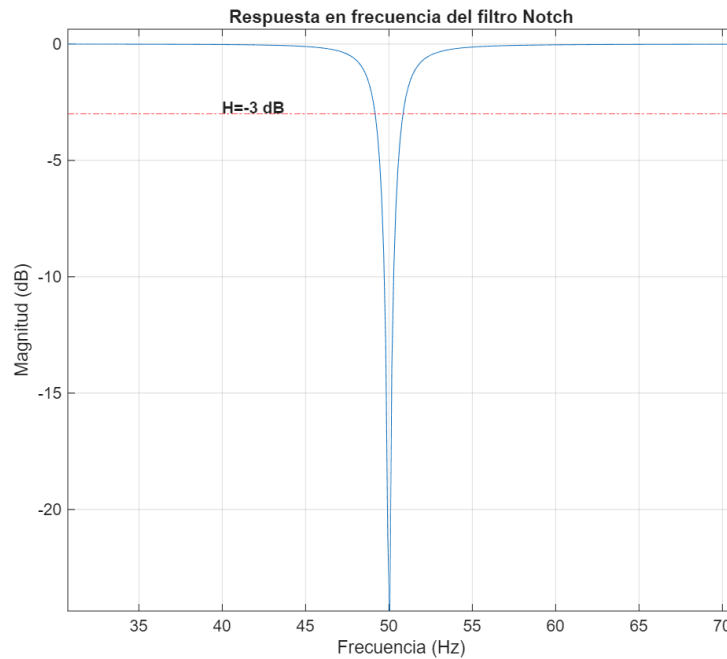


Figura 32: Respuesta en frecuencia del filtro Notch, centrado en la frecuencia normalizada

En la Figura 33 se muestra el efecto obtenido por el filtro Notch en la señal, donde puede apreciarse una ligera disminución del ruido, que será eliminado de manera más eficiente en las etapas posteriores del procesamiento. Por su parte, la Figura 34 presenta una comparativa de la Transformada Rápida de Fourier (FFT) de la señal obtenida en condiciones de reposo antes y después del filtrado. En dicha representación se observa una reducción evidente en la magnitud de la componente espectral correspondiente a 50 Hz, lo que confirma la efectividad del filtro en la atenuación de esta frecuencia indeseada.

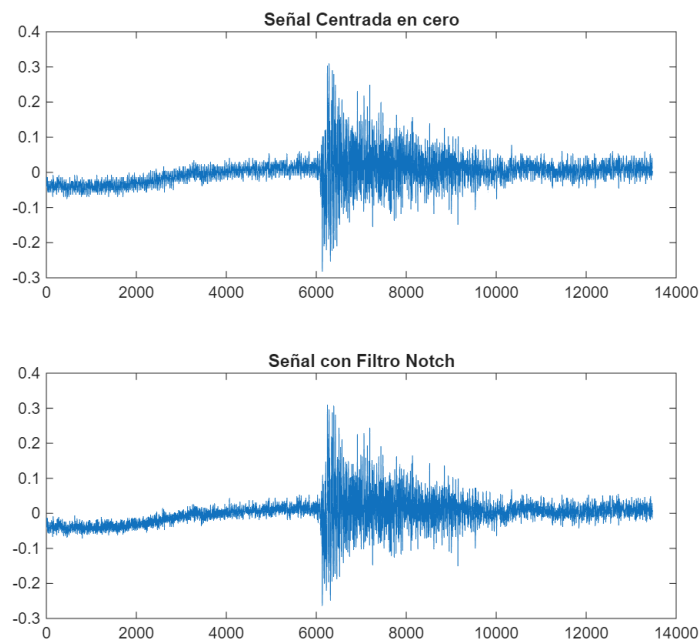


Figura 33: Comparación de la señal tras aplicar el filtro Notch

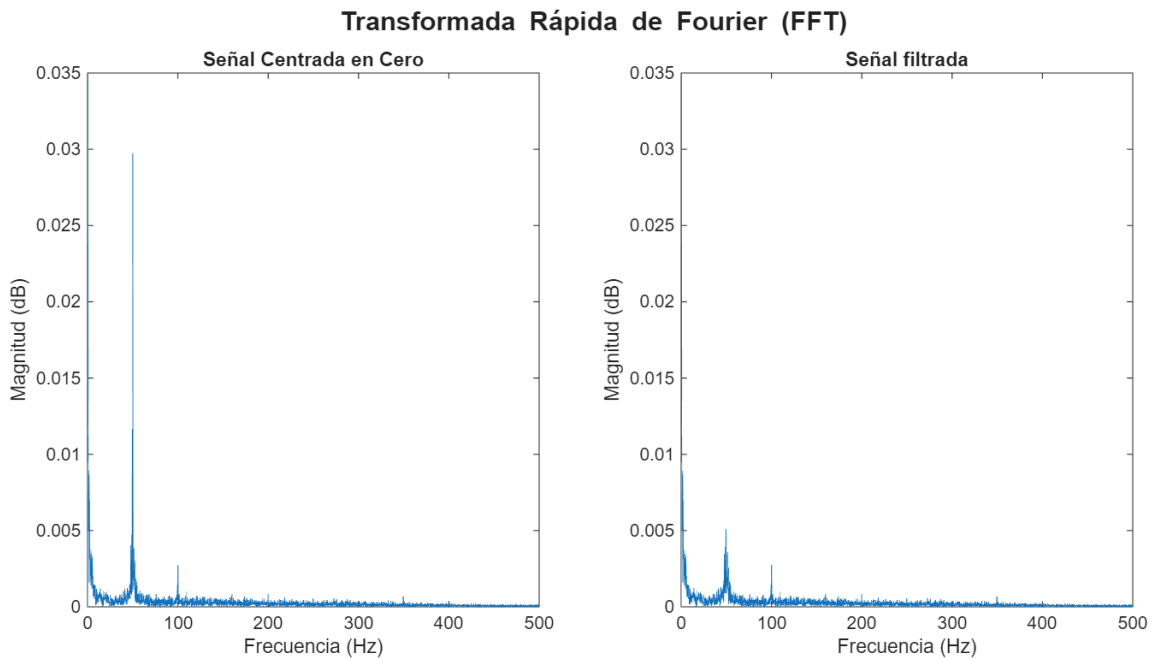


Figura 34: Comparativa de las FFT antes y después de aplicar el filtro Notch

5.3 Filtro paso banda

Una vez atenuado el ruido de la red eléctrica, se procederá a reducir lo máximo posible el ruido generado por componentes de baja y alta frecuencia que no aportan información relevante, sino que enmascaran la señal, impidiendo así su óptimo análisis. Para ello, se emplea un filtro paso banda, cuya función es permitir el paso de un rango específico de frecuencias de interés, mientras atenúa aquellas situadas por debajo y por encima de dicho rango (Carrasco, 2024).

Este tipo de filtros resulta imprescindible en el procesamiento de señales biomédicas, ya que permite aislar la banda espectral en la que se concentra la información significativa, eliminando artefactos de baja frecuencia producidos por movimiento y ruido de alta frecuencia proveniente del instrumental utilizado, así como de interferencias externas (Análisis de Señales Fisiológicas: Técnicas | StudySmarter, s. f.).

En este trabajo, el filtrado se realizó mediante un filtro paso banda Butterworth de quinto orden, seleccionado por su respuesta máximamente plana y sin ondulaciones en la banda de paso y de rechazo (Butterworth Filter - an Overview | ScienceDirect Topics, s.f.). Este filtro presenta una región de transición amplia entre la banda de paso y la de rechazo,

alejándose de la idealidad. Aumentar el orden haría la transición más abrupta y selectiva, pero también podría distorsionar la señal y alterar parte de la información de interés.

Otros filtros, como los de Chebyshev, permiten lograr transiciones más rápidas, aunque a cambio introducen ondulaciones que también pueden afectar a la señal (Butterworth Filter - an Overview | ScienceDirect Topics, s.f.). En este caso, el filtro Butterworth de orden 5 fue suficiente para reducir el ruido sin comprometer la calidad de la señal procesada.

Para la implementación del filtro se empleó la función *butter* de MATLAB, especificando como parámetros las frecuencias de corte inferior y superior, la frecuencia de muestreo y el orden del filtro, con el objetivo de obtener los coeficientes correspondientes. Se establecieron como frecuencias de corte 50 Hz y 150 Hz, dado que en dicho intervalo espectral se concentra la mayor parte de la energía de la señal.

El resultado de la aplicación del filtro a la señal, tras el preprocesamiento inicial, se muestra en la Figura 35, donde se presenta una comparativa entre la señal sin apenas procesamiento, la señal corregida del componente de continua y la señal filtrada mediante un paso banda. En dicha representación se aprecia una reducción de la amplitud de varias décimas, lo que evidencia la atenuación del ruido y una mayor limpieza del registro.

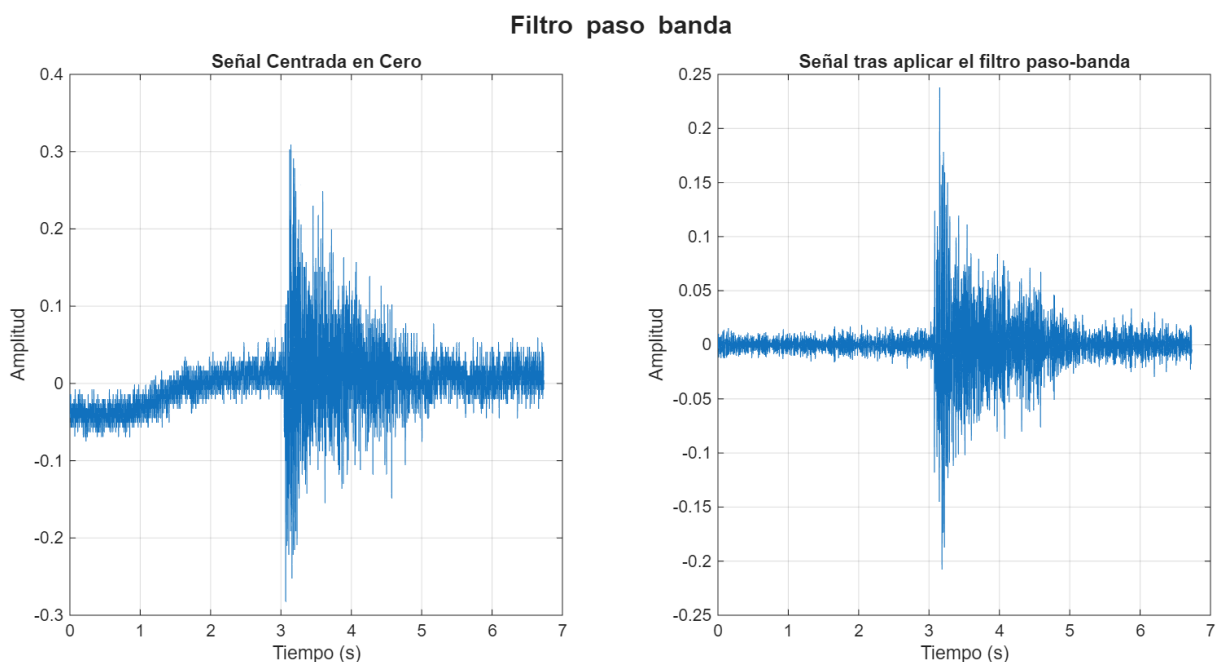


Figura 35: Comparativa de la señal antes y después del filtro paso banda

6

Extracción de características

Tras el procesamiento de las señales y la eliminación del ruido no deseado, se procede a la etapa de análisis de las señales electromiográficas, tanto en el dominio temporal como en el frecuencial. El objetivo de esta fase es transformar la señal cruda en un conjunto de parámetros cuantitativos que permitan describir de manera eficiente la actividad muscular y diferenciar los distintos gestos incluidos en el estudio.

La extracción de características constituye un paso esencial en el procesamiento de señales EMG, ya que reduce la complejidad de los datos originales y resalta aquellas propiedades más representativas para la posterior clasificación de los gestos. Para ello, se seleccionaron un conjunto de descriptores comúnmente empleados en la literatura, distribuidos en los dos dominios principales de análisis.

6.1 Dominio temporal

Las características del dominio temporal se basan en parámetros estadísticos que describen la forma de onda de la señal, tales como la media o la varianza. Estas medidas permiten analizar la amplitud y la energía de la señal en intervalos concretos de tiempo, ofreciendo información sobre el nivel de activación muscular, el tono y la posible aparición de fatiga. Gracias a su simplicidad computacional, resultan especialmente útiles en aplicaciones en tiempo real, como el control de prótesis mioeléctricas, donde es fundamental una extracción rápida y eficiente de la información (Moctar et al., 2023).

Existen múltiples características citadas en la literatura para el reconocimiento de gestos, de las cuales para este trabajo se seleccionaron las siguientes (Comparison Classification Of EMG Signals In The Time Domain And Time-Frequency Domain, 2023; Signal Features - MATLAB & Simulink, 2025):

1. **Media aritmética** (Mean): representa el valor promedio de la señal durante un período determinado de tiempo e indica el valor medio de la amplitud de la señal, utilizado en mayor medida como control de calidad de la línea base de la señal.
2. **Media cuadrática** (*Root Mean Square, RMS*): es una de las características más representativas, ya que refleja la intensidad de la actividad muscular. El RMS permite diferenciar de forma eficiente distintos gestos o patrones de activación, siendo además proporcional a la fuerza generada durante contracciones isométricas. Por otro lado, es un buen indicativo de la fatiga muscular (Papandrea, 2024).
3. **Desviación estándar** (Standard Deviation): mide la dispersión de los valores respecto a la media. En señales EMG, donde la media suele ser aproximadamente 0, la desviación estándar es prácticamente equivalente al RMS. La desviación estándar refleja la variabilidad de la amplitud de la señal y aporta información sobre el grado de activación muscular.
4. **Factor de forma** (Shape Factor): es la relación entre el RMS y el valor medio absoluto (MAV) de la señal. Permite caracterizar la forma de la onda y diferenciar contracciones musculares según la regularidad de su actividad eléctrica.

5. **Relación señal-ruido** (Signal-to-Noise Ratio, SNR): es la relación entre la potencia útil de la señal y la potencia del ruido. Se utiliza para evaluar la calidad de la señal y se expresa en decibelios.
6. **Distorsión armónica total** (Total Harmonic Distortion, THD): es la relación entre la potencia útil de la señal y la potencia de la distorsión. Cuantifica la presencia de armónicos respecto al componente fundamental. Refleja la distorsión eléctrica de la señal, actuando, así como un indicador más de la calidad de la señal. Se expresa en decibelios.
7. **Relación señal-ruido y distorsión** (Signal to Noise and Distorsion Ratio, SINAD): es la relación entre la potencia útil de la señal y la suma de la potencia del ruido y de la distorsión. Se utiliza para evaluar la calidad de la señal y se expresa en decibelios.
8. **Valor pico** (Peak Value): corresponde a la máxima amplitud alcanzada por la señal en un intervalo de tiempo. Es sensible a contracciones rápidas y bruscas, pero se utiliza para el cálculo de otras medidas, tales como el factor de cresta o impulso.
9. **Factor de cresta** (Crest Factor): es la relación entre el valor pico y el RMS. Indica el grado de fluctuación de una señal, permitiendo distinguir contracciones musculares estables de aquellas con gran variabilidad o picos aislados.
10. **Factor de despeje** (Clearence Factor): se define como el valor pico dividido por la media de las raíces cuadradas de las amplitudes absolutas. Permite detectar la presencia de picos transitorios y caracterizar la irregularidad de la contracción muscular, siendo los valores elevados un indicativo de señales con descargas bruscas o artefactos, mientras que valores bajos reflejan una actividad más uniforme y estable.
11. **Factor de impulso** (Impulse Factor): Es la relación entre el valor pico y el valor medio absoluto de la señal. Mide el grado de impulsividad de la señal, ayudando en la identificación de artefactos de movimiento e interferencias. Valores altos indican que la señal está dominada por movimientos bruscos o artefactos, mientras que valores bajos representan una contracción sostenida y estable.

En suma, las características del dominio temporal ofrecen una base sólida para el análisis de la señal EMG, al integrar medidas que reflejan activación, estabilidad y calidad. Esta combinación permite obtener una visión global del comportamiento muscular y refuerza su aplicabilidad en ámbitos clínicos y tecnológicos.

6.2 Dominio frecuencial

El análisis de las señales electromiográficas en el dominio frecuencial se basa en la descomposición de la señal en sus componentes de frecuencia, permitiendo identificar patrones que no son evidentes en el dominio temporal. Las características frecuenciales, tales como la frecuencia media o mediana, reflejan la distribución de la energía de la señal en distintas bandas de frecuencia y, aunque su coste computacional es más elevado que las del dominio temporal, permiten obtener información útil acerca de la fatiga y fuerza muscular (Too et al., 2017).

Al igual que en el dominio temporal, existen múltiples características empleadas en estudios de reconocimiento de gestos a partir de señales EMG, de entre las cuales, para este trabajo se han utilizado las siguientes:

12. **Frecuencia media** (Mean Frequency, MeanFreq): representa el promedio ponderado de todas las frecuencias presentes en el espectro de la señal EMG. Se utiliza como indicador del contenido espectral global y está estrechamente asociado a la activación muscular y a la detección de fatiga, ya que esta última suele producir un desplazamiento hacia frecuencias más bajas.
13. **Frecuencia mediana** (Median Frequency, MedFreq): corresponde al valor de frecuencia que divide el espectro de potencia en dos áreas iguales. Es uno de los parámetros más empleados para monitorizar la fatiga muscular, dado que tiende a disminuir de manera progresiva con la prolongación de la contracción.
14. **Potencia de banda** (Band Power): mide la energía contenida en una banda específica del espectro de frecuencias. Permite analizar la contribución de diferentes rangos de frecuencia a la señal EMG, lo que resulta útil para identificar patrones de activación asociados a determinados grupos musculares o gestos.

15. **Ancho de banda ocupado** (Occupied Bandwidth): define el rango de frecuencias en el cual se concentra un porcentaje determinado de la potencia total de la señal (habitualmente el 99%). Este parámetro refleja la extensión espectral de la actividad muscular y ayuda a diferenciar contracciones regulares de aquellas más complejas o irregulares.
16. **Ancho de banda de potencia** (Power Bandwidth): se refiere al intervalo de frecuencias delimitado por la amplitud en decibelios (generalmente -3 dB) respecto al valor máximo del espectro. Permite caracterizar la dispersión de la energía espectral y evaluar la selectividad de la señal.
17. **Amplitud pico** (Peak Amplitude): corresponde al valor máximo de amplitud del espectro de la señal. Este indicador refleja la intensidad dominante en frecuencia y se emplea para analizar la potencia máxima asociada a la activación muscular.
18. **Ubicación del pico espectral** (Peak Location): identifica la frecuencia exacta en la que se localiza la amplitud máxima del espectro. Su análisis permite detectar la frecuencia dominante de la señal, lo que proporciona información directa sobre la condición fisiológica del músculo y la posible aparición de fatiga.

La suma de las características de ambos dominios permite obtener un análisis completo de las señales, facilitando el trabajo de reconocimiento de patrones al clasificador posterior. Mientras que las características temporales destacan por su simplicidad y rapidez en aplicaciones en tiempo real, las frecuenciales aportan una visión más detallada sobre la distribución espectral y los cambios fisiológicos asociados a la actividad muscular.

Para la obtención de las características se emplearon las funciones *signalTimeFeatureExtractor* (*signalTimeFeatureExtractor* - Streamline Signal Time Feature Extraction - MATLAB, s. f.) y *signalFrequencyFeatureExtractor* (*signalFrequencyFeatureExtractor* - Streamline Signal Frequency Feature Extraction - MATLAB, s. f.) de MATLAB, junto a la función *extract*, que generaron como resultado una tabla con todas las características de tiempo y frecuencia extraídas por cada canal y movimiento como ilustra la Figura 36, lista para ser utilizada como entrada de un clasificador basado en inteligencia artificial. Este clasificador será el encargado de analizar los patrones presentes en los

descriptores extraídos y distinguir de manera automática entre los diferentes gestos realizados.

		1	2	3	4	5
		channel1_Mean	channel1_RMS	channel1_StandardDeviation	channel1_ShapeFactor	channel1_SN
1	abrir_u1_rep1	-2.1449e-06	0.0222	0.0222	1.8403	-
2	abrir_u1_rep10	9.3081e-06	0.0236	0.0236	1.6620	-1
3	abrir_u1_rep2	2.0755e-07	0.0185	0.0185	1.7047	-1
4	abrir_u1_rep3	1.0330e-06	0.0208	0.0208	1.8042	-1
5	abrir_u1_rep4	9.3886e-07	0.0244	0.0244	1.7098	-1
6	abrir_u1_rep5	3.2511e-06	0.0180	0.0180	1.6536	-1
7	abrir_u1_rep6	-3.0373e-06	0.0187	0.0187	1.7835	-1
8	abrir_u1_rep7	-4.4820e-06	0.0260	0.0260	1.8508	-1
9	abrir_u1_rep8	6.2631e-07	0.0195	0.0195	1.6418	-1
10	abrir_u1_rep9	4.9130e-06	0.0226	0.0226	1.6154	-1
11	abrir_u2_rep1	9.0320e-07	0.0214	0.0214	1.6507	-
12	abrir_u2_rep10	1.5175e-06	0.0158	0.0158	1.3970	-
13	abrir_u2_rep2	5.4490e-06	0.0157	0.0157	1.4136	-
14	abrir_u2_rep3	2.6449e-05	0.0180	0.0180	1.4428	-1
15	abrir_u2_rep4	2.5629e-06	0.0153	0.0153	1.4598	-
16	abrir_u2_rep5	-2.4278e-06	0.0147	0.0147	1.5048	-
17	abrir_u2_rep6	-4.6265e-07	0.0146	0.0146	1.3770	-

Figura 36: Tabla de características temporales y frecuenciales extraídas por canal y movimiento.

7

Clasificador de gestos

En esta última fase del reconocimiento de gestos mediante señales EMG se emplearán las características temporales y frecuenciales previamente extraídas para entrenar un clasificador de gestos, implementando técnicas de aprendizaje automático que permitan establecer correspondencias entre los descriptores extraídos y los movimientos musculares asociados. El objetivo es dotar al sistema de la capacidad de reconocer con precisión cada gesto para, finalmente, posibilitar el control de un brazo robótico mediante señales EMG, configurando una interfaz hombre-máquina robusta y funcional, aunque orientada a un procesamiento offline o semi-online.

7.1 Estandarización de los datos

Previa a la implementación del clasificador y, dado que las características extraídas presentan diferentes unidades (dB, adimensional, etc.) y escalas, será necesaria la estandarización de los datos.

Para ello, se utilizará la estandarización o normalización Z-score, que consiste en la transformación de los datos de manera que tengan una media de 0 y una desviación estándar de 1. De esta forma, se consigue que todas las métricas se encuentren en el mismo rango de valores y con la misma unidad, simplificando así el procesamiento de los datos y aumentando

la precisión de los clasificadores (Manu Duque, 2025). La estandarización se lleva a cabo mediante la siguiente fórmula:

$$z_i = \frac{(x_i - \mu)}{\sigma} \quad (7)$$

donde z_i es el valor del dato estandarizado, x_i es el valor original del dato previo a la normalización, μ es la media del conjunto de datos y σ es la desviación estándar, que indica la separación de los datos con respecto a la media.

La estandarización se llevará a cabo mediante la función *zscore* de MATLAB, y se guardarán los valores de σ y μ para ser aplicados a los datos de prueba, una vez el clasificador sea entrenado.

7.2 Implementación del clasificador

Una vez se haya realizado la estandarización de las señales, de manera que todas se encuentren dentro del mismo rango de valores, se implementará el clasificador de gestos.

Para ello, se utilizará la aplicación *Classification Learner* de MATLAB, que ofrece una interfaz gráfica muy visual y sencilla para entrenar y comparar múltiples modelos de clasificación sobre el mismo conjunto de datos. Además, esta herramienta permite seleccionar el esquema de validación, los predictores que se usarán para el entrenamiento, la clase de salida e incluso ajustar los hiperparámetros de cada modelo (Classification Learner - Train Models To Classify Data Using Supervised Machine Learning - MATLAB, s. f.).

En este trabajo se entrenarán y evaluarán distintos algoritmos de clasificación disponibles en la aplicación con el objetivo de identificar cuál ofrece una mayor precisión en la detección de los gestos. Entre los modelos considerados se incluyen: árboles de decisión, modelos Naive Bayes, máquinas de vectores de soporte (SVM), vecinos más cercanos (KNN), redes neuronales, análisis discriminante lineal, regresión logística y modelos de ensamble. Cada uno de estos clasificadores será probado con diferentes configuraciones de hiperparámetros con la finalidad de elegir el que mejor se adapte a los datos de entrenamiento y ofrezca una mayor precisión y robustez en la clasificación.

En la Figura 37 se muestra la configuración inicial escogida, donde se usará la validación cruzada con 10 pliegues (*10-fold cross validation*). En este procedimiento, el conjunto de entrenamiento se divide en 10 partes iguales (*folds*), y el modelo se entrena 10 veces, de manera que en cada iteración se utilizan 9 pliegues como conjunto de entrenamiento y el pliegue restante como conjunto de validación. De esta forma, cada pliegue actúa una vez como validación y nueve veces como entrenamiento.

Finalmente, se obtiene un valor medio de precisión a partir de los 10 resultados obtenidos, lo que proporciona una estimación más robusta y fiable del rendimiento real del modelo, minimizando el riesgo de que los resultados dependan de una única partición aleatoria de los datos.

Data set

Data Set Variable: standardized_features (280x145 table)

Response: From data set variable, From workspace. Selected: Movimiento (cell, 7 unique)

Predictors

	Name	Type	Range
<input checked="" type="checkbox"/>	channel1_Mean	double	-6.57396 .. 6.45825
<input checked="" type="checkbox"/>	channel1_RMS	double	-1.45591 .. 4.85923
<input checked="" type="checkbox"/>	channel1_StandardDeviation	double	-1.45591 .. 4.85923
<input checked="" type="checkbox"/>	channel1_ShapeFactor	double	-1.25328 .. 6.33172
<input checked="" type="checkbox"/>	channel1_SNR	double	-1.59555 .. 4.50407
<input checked="" type="checkbox"/>	channel1_THD	double	-2.73552 .. 1.78944

Buttons: Add All, Remove All, Refresh

[How to prepare data](#)

Validation

Validation Scheme: Cross-Validation

Protects against overfitting. For data not set aside for testing, the app partitions the data into folds and estimates the accuracy on each fold.

Cross-validation folds: 10

[Read about validation](#)

Test

Set aside a test data set

Percent set aside: 10

Use a test set to evaluate model performance after tuning and training models. To import a separate test set instead of partitioning the current data set, use the Test Data button after starting an app session.

[Read about test data](#)

Buttons: Start Session, Cancel

Figura 37: Configuración inicial de parámetros en la aplicación Classification Learner

7.3 Comparación de modelos

Dado que los modelos empleados en el entrenamiento han mostrado resultados de exactitud similares, se llevará a cabo una comparación entre los cuatro con mayor porcentaje

de acierto basada en métricas como la curva ROC o el área bajo la curva (AUC). Posteriormente, se realizarán pruebas específicas con cada uno de ellos con el fin de identificar y analizar las diferencias en su rendimiento.

La curva ROC (*Receiver Operating Characteristic*) es una representación gráfica del rendimiento de los modelos de clasificación que representa la relación entre la tasa de falsos positivos (FPR) en el eje X y la tasa de verdaderos positivos (TPR) en el eje Y. Cada punto de la curva corresponde a un umbral de decisión distinto utilizado por el modelo para clasificar las observaciones (Sanahuja, s. f.).

Cuanto más se acerque la curva a la esquina superior izquierda del gráfico, mejor será el rendimiento del modelo, ya que significa que alcanza una alta tasa de verdaderos positivos con una baja tasa de falsos positivos.

Por su parte, el área bajo la curva (AUC) es una medida complementaria a la curva ROC, ya que cuantifica el área comprendida bajo dicha curva, asignando un valor entre 0 y 1. De este modo, cuanto más ideal sea la curva, es decir, cuanto más próxima se sitúe de la esquina superior izquierda del gráfico, mayor será el valor de AUC, lo que indica un mejor rendimiento del modelo de clasificación (Sanahuja, s. f.). Por el contrario, un valor igual o inferior a 0.5, indica que la capacidad predictiva del modelo es similar o incluso peor que la de una clasificación aleatoria.

Conociendo esta información, se graficarán las curvas ROC y el AUC de cada uno de los 4 modelos seleccionados: modelo 27.3-Ensemble Subspace Discriminant (89.6%), modelo 11-Quadratic SVM (89.3%), modelo 12-Cubic SVM (88.9%) y modelo 30-Wide Neural Network (88.2%).

Como se muestra en la Figura 38, la curva ROC más cercana a la idealidad es la del modelo de ensamble, con un AUC de 0.9978, prácticamente 1. A este le siguen los modelos de máquina de vectores de soporte, siendo el cúbico el que presenta un mejor rendimiento (AUC=0.9947) frente al cuadrático (AUC=0.9931). Por último, el modelo de red neuronal presenta el rendimiento más bajo, con una menor tasa de verdaderos positivos y un AUC=0.9815.

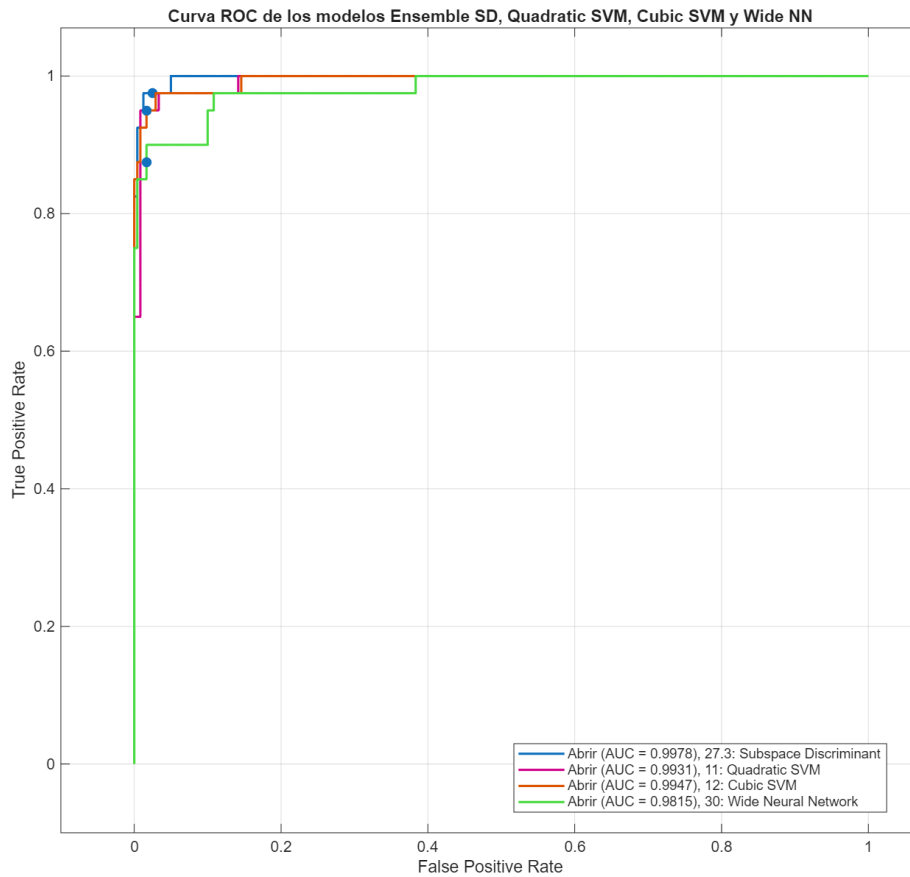


Figura 38: Comparación de las curvas ROC de los diferentes modelos

Por otro lado, al analizar las gráficas de la Tabla 4, en las que se comparan la exactitud, la tasa de error y el coste computacional de los cuatro modelos, se observa que el modelo de ensemble es el que ofrece un rendimiento superior.

Del mismo modo, al clasificar datos de prueba con los cuatro clasificadores se obtuvieron resultados semejantes, aunque destacó la eficacia del modelo de ensemble. En este caso, se evaluaron 21 datos de prueba, 3 por cada tipo de movimiento, de los cuales se lograron 16 aciertos utilizando dicho modelo. Estos resultados se presentan en la Tabla 5.

En consecuencia, el modelo seleccionado para llevar a cabo la clasificación es el Ensemble Subspace Discriminant.

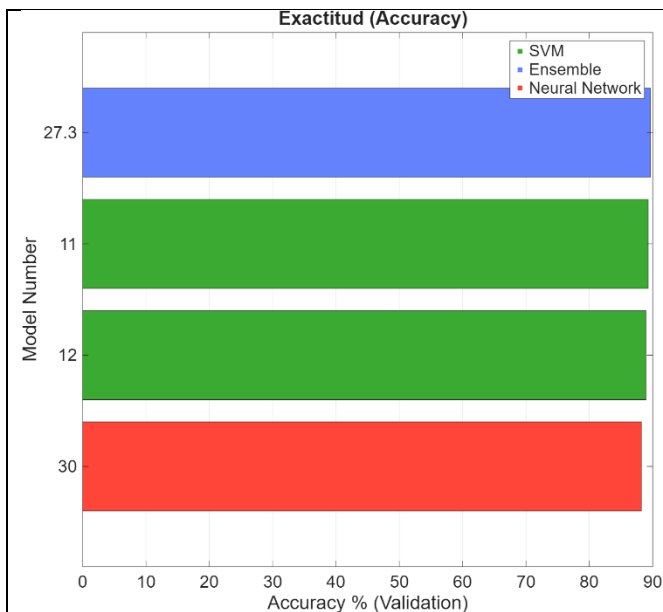


Figura 39: Comparación de la exactitud de los modelos

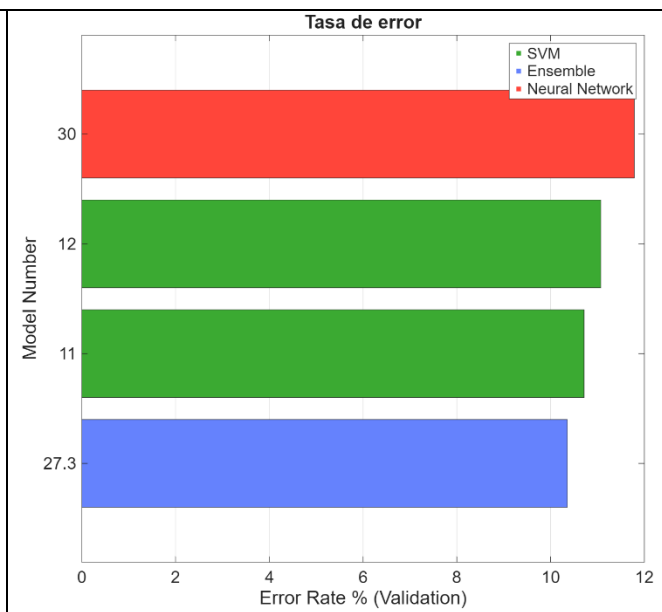


Figura 40: Comparación de la tasa de error de los modelos

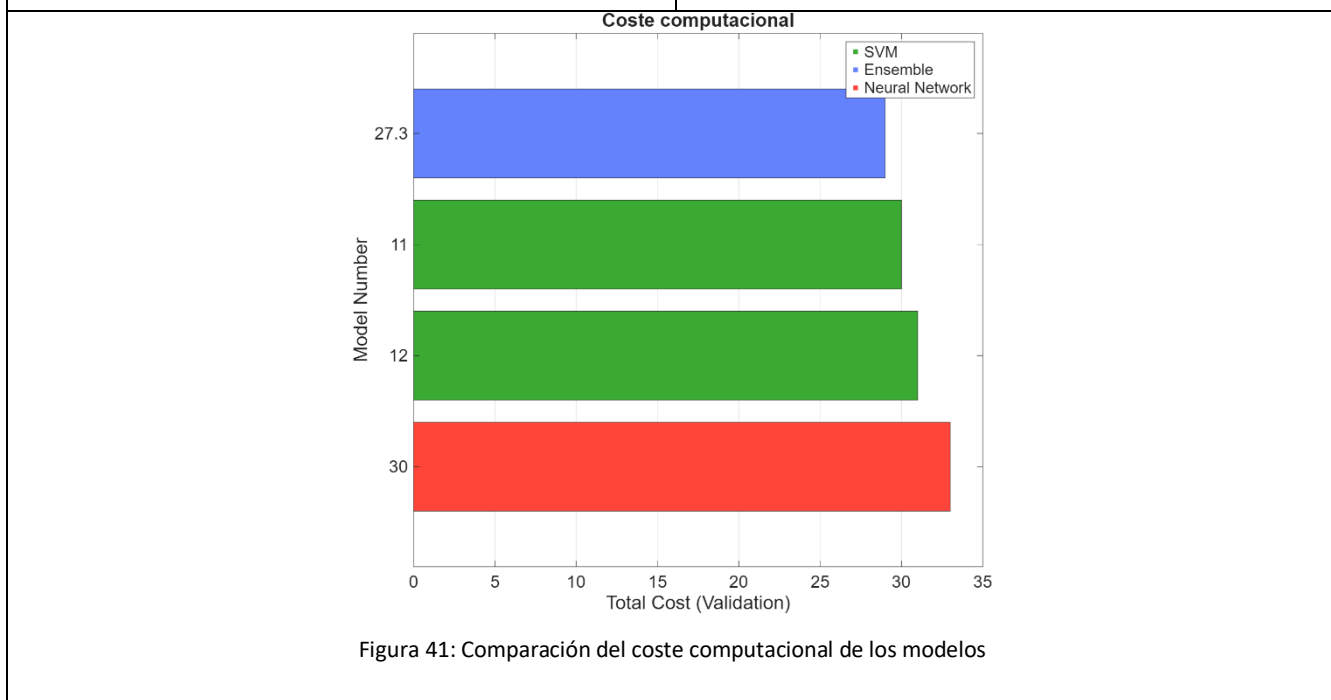


Figura 41: Comparación del coste computacional de los modelos

Tabla 4: Comparación de la exactitud, tasa de error y coste computacional de los diferentes modelos

Modelo Movimiento	Ensemble Subspace Discriminant	Quadratic SVM	Cubic SVM	Wide Neural Network
	Aciertos			
Abrir	1/3	1/3	1/3	1/3
Cerrar	2/3	2/3	2/3	2/3
Pronación	3/3	3/3	2/3	1/3
Supinación	3/3	2/3	2/3	2/3
Reposo	3/3	3/3	3/3	3/3
Pulgar	1/3	1/3	1/3	2/3
Dos dedos	3/3	3/3	3/3	3/3
	16/21	15/21	14/21	14/21

Tabla 5: Comparación de los resultados de las pruebas de los clasificadores

7.4 Resultados del entrenamiento

De entre todos los modelos entrenados, el que mayor exactitud ofrece es el Ensemble Subspace Discriminant, con un 89.6%. Este modelo se basa en los modelos de ensamble, que combinan las propiedades de diferentes clasificadores no muy potentes para crear un clasificador completo y con mejor rendimiento, de manera que los posibles errores de cada uno de ellos queden contrarrestados al fusionarlos (Modelos de Ensamble (Ensemble Models) | AI Planet (Formerly DPhi), s. f.).

Por otro lado, el término *subspace* se refiere a que cada clasificador base dentro del ensamble no utiliza todas las variables de entrada, sino un subconjunto aleatorio de características. De esta forma, cada clasificador base es entrenado por variables diferentes, lo que introduce diversidad en el conjunto de clasificadores y evita que todos cometan los mismos errores (Ashour et al., 2018).

En cuanto al término *discriminant*, este hace referencia a que el clasificador base empleado en cada subespacio es un análisis discriminante lineal (LDA). Este método busca encontrar combinaciones lineales de las variables que permitan separar de la mejor manera posible las distintas clases (Ibm, 2025). Aunque de forma individual el discriminante lineal es

un modelo relativamente sencillo y puede no ser muy potente, al integrarse dentro del ensamble gana en capacidad predictiva gracias a la diversidad de los subespacios de características y a la combinación de múltiples votos.

En este caso, el modelo completo cuenta con 30 clasificadores base, donde cada uno es entrenado por 72 predictores seleccionados al azar del total disponible, como muestra la Figura 42. Cuando se introduce un dato de prueba o validación, cada clasificador realiza una predicción sobre la clase a la que pertenece. Finalmente, la clase final será el resultado de una mayoría absoluta de entre todas las predicciones.

Gracias a este enfoque, el Ensemble Subspace Discriminant consigue potenciar la eficacia de cada uno de los clasificadores, logrando elevar la exactitud del modelo al 89.6 % con un bajo coste computacional y una baja tasa de error.

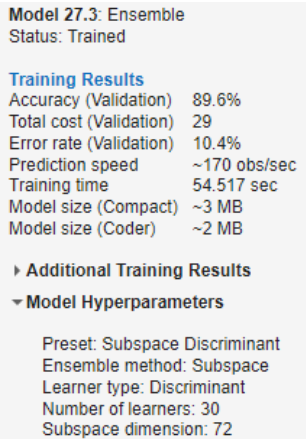


Figura 42: Resultados del entrenamiento del Ensemble Subspace Discriminant

En la Figura 43 se presenta la matriz de confusión correspondiente al modelo seleccionado. En ella se incluyen, además, las tasas de verdaderos positivos (*True Positive Rate*, TPR) y de falsos negativos (*False Negative Rate*, FNR).

La matriz de confusión es una herramienta fundamental para evaluar el rendimiento del modelo de aprendizaje automático, ya que permite comparar los valores reales con los valores predichos por el modelo. En este caso, el eje horizontal representa los valores reales y el eje vertical la predicción del modelo (Arce, 2024).

Las celdas situadas en la diagonal principal indican las coincidencias entre la clase verdadera y la predicha, es decir, los aciertos del modelo o verdaderos positivos, mientras que los valores fuera de esta representan los fallos, que pueden corresponder a falsos positivos o falsos negativos (Visualize And Assess Classifier Performance In Classification Learner - MATLAB & Simulink, s. f.). Los falsos positivos se refieren a casos en los que el modelo predice erróneamente una clase negativa como positiva, mientras que los falsos negativos aparecen cuando el modelo no detecta la clase positiva y la clasifica de manera incorrecta como negativa.

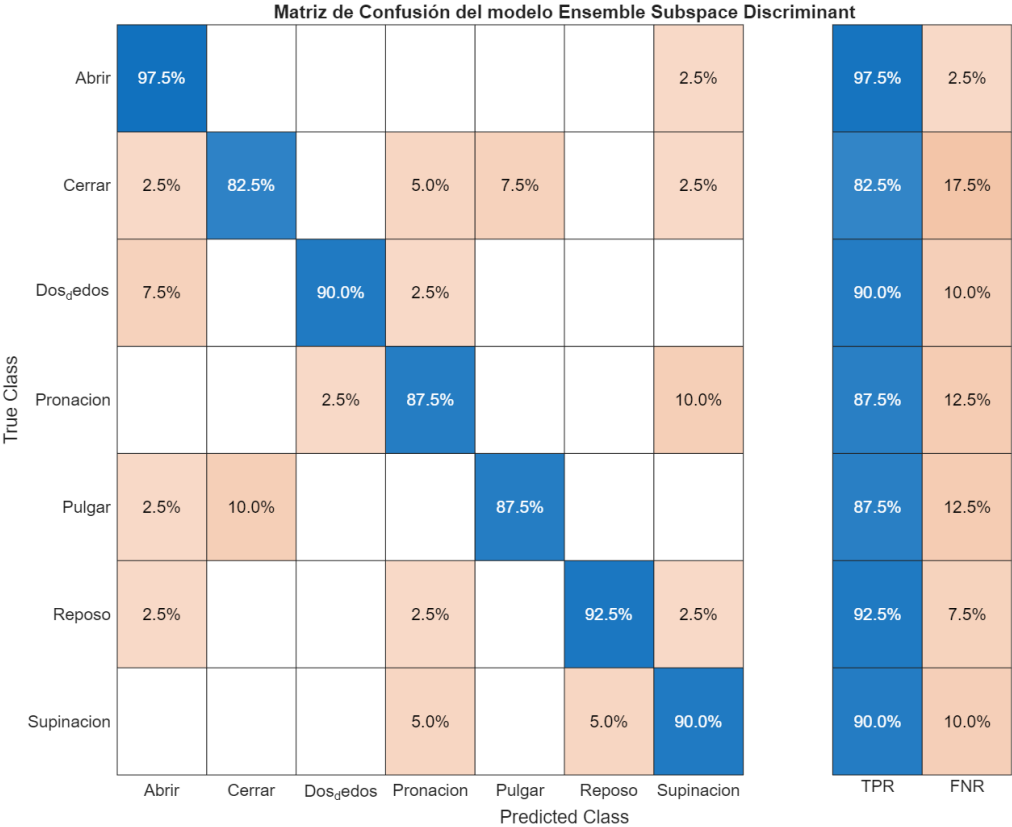


Figura 43: Matriz de confusión, TPR y FNR del modelo elegido

Al analizar la matriz, se observa que el modelo presenta en general un alto rendimiento, con tasas de acierto superiores al 85% en la mayoría de las clases. El movimiento con mejor desempeño corresponde a la apertura de mano, con una TPR de 97.5%, lo que indica una excelente capacidad del modelo para reconocer este gesto.

Por el contrario, el movimiento que presenta un mayor número de errores de clasificación es el de cierre de mano, con un FNR de 17.5%. Este gesto se confunde principalmente con la extensión de pulgar (7.5%), dado que ambos movimientos presentan gran similitud, diferenciándose únicamente en la flexión o no del pulgar.

Finalmente, los gestos de pronación y supinación también muestran confusiones entre sí, dado que ambos movimientos presentan un cierto grado de similitud biomecánica.

En suma, el modelo *Ensemble Subspace Discriminant* demuestra un desempeño robusto y adecuado para la clasificación de los distintos gestos, alcanzando altos valores de TPR en casi todas las clases. No obstante, se identifican ciertas limitaciones en la discriminación de movimientos con gran proximidad biomecánica (Cierre de mano–Extensión de pulgar y Pronación–Supinación), aunque estos errores no limitan el potencial del clasificador.

8

Control del brazo robótico

En este último capítulo se presenta la aplicación práctica del clasificador de gestos desarrollado en el capítulo anterior, integrándolo en el control de un brazo robótico impreso en 3D. Este dispositivo, cuyos actuadores se basan en servomotores y cables, se plantea como una posible aproximación para contribuir a mejorar la calidad de vida de las personas amputadas. En un contexto donde muchas de las prótesis existentes resultan costosas, pesadas o con un control limitado, esta propuesta busca ofrecer una alternativa mioeléctrica experimental orientada a recuperar, al menos en parte, la movilidad perdida de manera más accesible e intuitiva.

Para ello se empleará el brazo desarrollado en el Proyecto de Fin de Carrera “Montaje y control de un brazo robótico impreso en 3D”, realizado por Alberto José Tudela Roldán, graduado en Ingeniería de Telecomunicaciones por la Universidad de Málaga, en el año 2015 (Tudela, 2015).

8.1 Estructura, control y alimentación del brazo robótico

El brazo formaba parte de un robot constituido por dos extremidades superiores completas, desde los dedos hasta el hombro, tal y como se muestra en la Figura 44, donde falta el brazo en cuestión. El robot completo fue diseñado y desarrollado en impresión 3D, utilizando servomotores de diferente potencia y cables rígidos, que actúan como tendones, para permitir el movimiento de sus articulaciones.



Figura 44: Robot completo

Tal como se aprecia en la Figura 45, el brazo está constituido por dos elementos principales: el antebrazo, que alberga los servomotores encargados de accionar los dedos mediante sus respectivos tendones, y la mano, que se acopla al primero. En particular, los servomotores de los dedos permiten su flexión y extensión al girar hasta 180° , reproduciendo así el movimiento básico de cierre y apertura de la mano. Ambos componentes se articulan a través de un sistema de engranajes conectado a un servomotor, el cual permite realizar un giro de muñeca de hasta 180° .

El control de los servomotores se lleva a cabo mediante un microcontrolador Arduino Nano, que actúa como núcleo del sistema. Este dispositivo, al estar enlazado con el ordenador, recibe y ejecuta las órdenes programadas en la plataforma Arduino, coordinando de forma precisa los movimientos del brazo robótico.

Por otra parte, el microcontrolador se alimenta mediante una fuente de laboratorio configurada en paralelo entre sus dos canales, ya que con un único canal la corriente no era suficiente y los servomotores no podían funcionar todos al mismo tiempo. Con esta configuración, la fuente entrega aproximadamente 5,5 V y hasta 6 A, lo que permite el funcionamiento conjunto de los seis servomotores instalados: cinco HobbyKing HK15298B para el movimiento de los dedos (0,7–0,9 A cada uno) y un TowerPro MG996R para la muñeca (0,5–0,9 A). En total, la demanda máxima se sitúa en torno a 5,4 A, por lo que la fuente proporciona el margen necesario para garantizar un control estable y sin interrupciones.

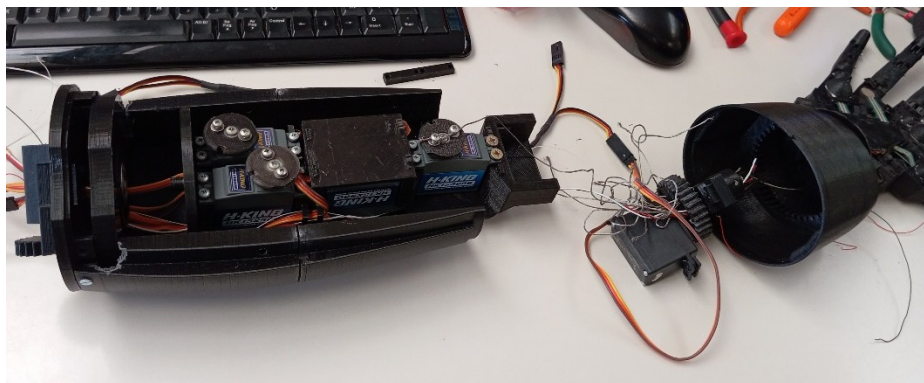


Figura 45: Componentes del brazo

8.2 Restauración y puesta en funcionamiento del brazo

Dado que el brazo fue desarrollado en 2015 y llevaba mucho tiempo sin utilizarse, fue necesario un ajuste y revisión previos antes de integrar el clasificador de gestos. Para ello, se llevaron a cabo una serie de acciones orientadas a restaurar su estado funcional:

1. Identificar a qué servomotor correspondía cada conector y qué dedo accionaba, en función de la posición de los servomotores en el antebrazo.

2. Verificar el correcto funcionamiento de los servomotores y sustituir aquellos que no respondían adecuadamente.
3. Tensar los tendones de los dedos y fijarlos a su servomotor correspondiente, tanto para la flexión como para la extensión.
4. Comprobar que el engranaje encajaba correctamente y permitía el movimiento de la muñeca.
5. Sustituir la pieza impresa en 3D que recubre la muñeca, ya que al atornillarla al antebrazo se había roto el filamento, quedando inestable. Para realizar el cambio fue necesario destensar de nuevo los tendones para poder extraer la pieza en mal estado. En la Figura 46 se puede apreciar el montaje de la nueva pieza al antebrazo, la cual encajó correctamente y devolvió firmeza al conjunto.
6. Ensamblar nuevamente el brazo completo y comprobar su funcionamiento.

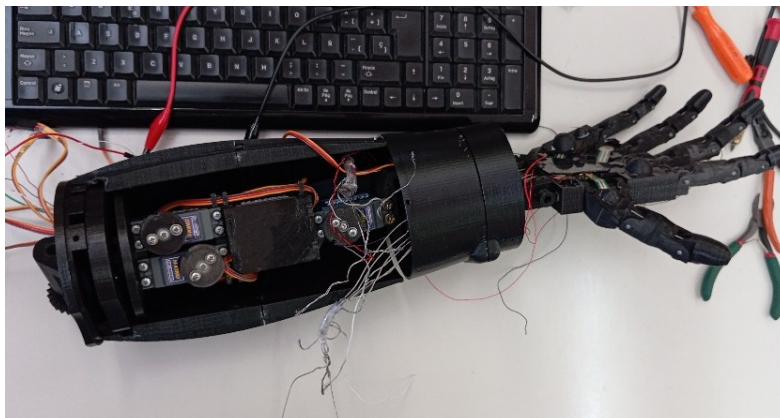


Figura 46: Montaje de la nueva pieza impresa de la muñeca

Finalmente, el brazo, ya restaurado y plenamente funcional, se muestra en la Figura 47.

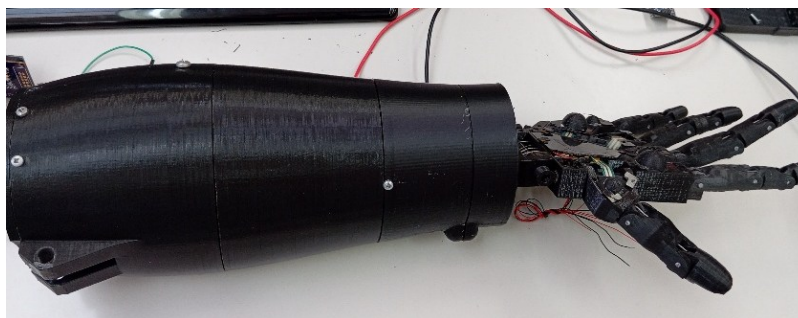


Figura 47: Brazo completo y restaurado

8.3 Integración del clasificador de gestos

La conexión entre el brazo robótico y el ordenador se establece a través de un puerto serie con el microcontrolador, a través del cual se intercambiarán datos a un baudrate de 115200. Para su control se utiliza el código de Arduino desarrollado por Alberto (Tudela, 2015), que permite manejar el robot mediante comandos sencillos. Con estas órdenes se pueden ejecutar tanto movimientos individuales como coordinados de los dedos y la muñeca.

Las instrucciones se organizan en tres niveles de complejidad, según el tipo de movimiento que permiten:

- Órdenes de tipo 1: ejecutan el movimiento sincronizado de todos los dedos para abrir o cerrar la mano. Se realizan mediante el comando O para abrir y C para cerrar.
- Órdenes de tipo 2: controlan de manera independiente un dedo o la muñeca, especificando el ángulo de giro del servomotor correspondiente. Los comandos son L, R, M, I, T y W, que indican los movimientos del meñique, anular, corazón, índice, pulgar y muñeca, respectivamente. Cada orden debe ir acompañada de un valor de ángulo entre 0 y 180, separados por coma.
- Órdenes de tipo 3: permiten mover varios dedos de forma coordinada, indicando el ángulo de giro deseado para cada uno. Se utilizan con el comando F, seguido de cinco valores comprendidos entre 0 y 180 (uno por dedo), separados por comas.

Con estas instrucciones, se elaboró una tabla que codifica el movimiento deseado en función de la salida del clasificador. De esta forma, la Tabla 6 muestra la correspondencia entre los gestos y los comandos necesarios para que el brazo robótico los ejecute.

En ella se establece que el gesto de reposo se representa moviendo los dedos a una posición medio flexionada y girando la muñeca hasta colocar la palma en posición lateral.

Los movimientos de pronación y supinación se definen como opuestos: en la pronación la muñeca se sitúa en 0°, de modo que la palma queda orientada hacia el suelo; mientras que en la supinación la palma se gira en sentido contrario, quedando orientada hacia el techo.

En cuanto a los gestos de extensión, la extensión del pulgar se logra flexionando los demás dedos a 180°, dejando únicamente el pulgar extendido. De manera similar, en la extensión del segundo y tercer dedos, todos los demás permanecen flexionados, manteniéndose extendidos únicamente el índice y el corazón.

Movimiento	Código
Abrir	“O”
Cerrar	“C”
Pronación	“W, 0”
Supinación	“W, 180”
Reposo	“F,90, 90, 90, 90, 90” “W,90”
Extensión de pulgar	“F,180, 180, 180, 180,0”
Extensión de segundo y tercer dedos	“F, 180, 180,0,0, 180”

Tabla 6: Codificación del movimiento del robot en función del gesto realizado

Para implementar la tabla se desarrolló una función en MATLAB encargada de enviar la información al microcontrolador del robot a través del puerto serie. Dicha función traduce cada gesto en los comandos correspondientes, según la codificación previamente descrita, garantizando así la correcta ejecución del movimiento por parte del brazo robótico.

Del mismo modo, se desarrollaron otras tres funciones adicionales: la primera destinada a la adquisición de las señales EMG; la segunda orientada al procesamiento y preparación de la información para su clasificación, aplicando los mismos parámetros empleados durante la estandarización de los datos de entrenamiento; y la tercera encargada de la clasificación utilizando el modelo seleccionado. La salida de esta última función define el gesto que finalmente ejecutará el robot.

En conjunto, estas cuatro funciones se integran en un script principal organizado en cascada, como muestra la Figura 48, de modo que la salida de cada módulo constituye la

entrada del siguiente, conformando una secuencia continua que culmina en el movimiento del brazo robótico.

```
% Abrir puerto para el brazo robótico
if ~exist('puerto_robot', 'var') || isempty(puerto_robot)
    puerto_robot = serialport("COM4",115200);
    configureTerminator(puerto_robot, "CR");
end

%Adquirir señal
leer_serie("COM3", 115200, 'pruebas.txt')

% Procesar señal
features = preprocesamiento(signal);

% Clasificar gesto
clase = clasificador(features);

% Mover el robot
movimiento_robot(clase, puerto_robot);
```

Figura 48: Script principal para el control del brazo

Para finalizar, tras la ejecución del script principal con el movimiento de extensión del pulgar, se obtuvieron las fotografías mostradas en la Figura 49, donde se aprecia el antes y el después de dicho movimiento. Esta figura evidencia tanto la eficacia del sistema de clasificación como la respuesta del brazo frente a la salida del clasificador, demostrando así su gran potencial como prototipo de prótesis.

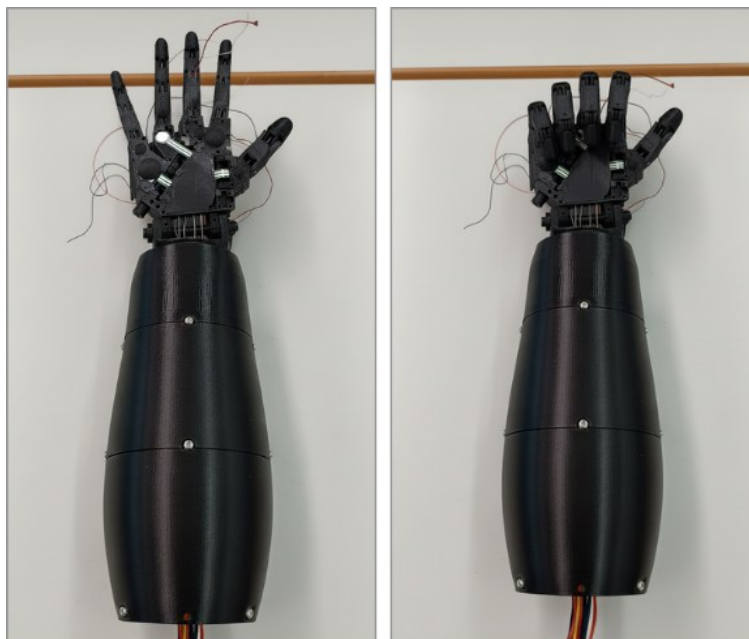


Figura 49: Respuesta del brazo a la extensión del pulgar obtenida como salida del clasificador

9

Conclusiones y líneas futuras

El presente Trabajo de Fin de Grado ha permitido demostrar la viabilidad de controlar un brazo robótico mediante el procesamiento de señales electromiográficas de superficie, suponiendo un avance clave en el camino hacia el desarrollo de prótesis mioeléctricas capaces de devolver autonomía y calidad de vida a las personas con amputaciones.

Gracias a la aplicación de diferentes técnicas de procesamiento, tales como la eliminación de la componente continua, la atenuación del ruido procedente de la red eléctrica y la reducción del ruido de alta y baja frecuencia, así como de los artefactos generados por movimiento, se ha conseguido obtener señales suficientemente limpias como para implementar un clasificador de gestos con resultados muy satisfactorios. En concreto, el sistema ha mostrado una capacidad de reconocimiento cercana al 76% de acierto en las pruebas realizadas, lo que representa un avance notable teniendo en cuenta la complejidad y variabilidad de las señales EMG.

Por otra parte, la utilización de MATLAB como entorno de trabajo ha facilitado la implementación de muchas de las funciones, como el extractor de características y el clasificador, gracias a sus herramientas complementarias. Además, se trata de un entorno accesible y bien documentado, con numerosos ejemplos prácticos que han contribuido a que el desarrollo fuera más ágil, eficiente y menos complejo.

De cara al futuro, será necesario seguir avanzando en la mejora del sistema, tanto en la capacidad de reconocimiento del clasificador de gestos como en la implementación en tiempo real, ya que el procesamiento de las señales introduce un tiempo de cómputo significativo. Por tanto, reducir este tiempo optimizará el flujo del sistema, permitiendo así acercar este prototipo a un uso práctico en prótesis.

Otra línea prometedora sería la posibilidad de incorporar señales electroencefalográficas (EEG) o potenciales evocados como complemento a las señales EMG. Estas señales cerebrales podrían aportar información adicional sobre la intención de movimiento y ayudar al sistema a clasificar de forma más fiable los gestos. Aunque son más complejas de registrar y procesar, su integración abriría la puerta a sistemas híbridos más precisos y adaptados a las necesidades reales de cada usuario.

En definitiva, este proyecto ha puesto de manifiesto el gran potencial de las señales biomédicas para el control de dispositivos robóticos. Aunque aún quedan retos por superar, los avances logrados sientan una base sólida para futuras investigaciones y acercan un poco más la posibilidad de desarrollar prótesis que respondan de forma natural a la intención de movimiento, contribuyendo a mejorar la autonomía y la calidad de vida de quienes las necesiten.

Bibliografía

- Tepe, C., & Demir, M. C. (2022). Real-Time Classification of EMG Myo Armband Data Using Support Vector Machine. *IRBM*, 43(4), 300–308. <https://doi.org/10.1016/j.irbm.2022.06.001>
- Ibarra-Fuentes, A., Vega-Espitia, A., Salazar-Martínez, J. L., Salcedo-Ramírez, J. L., Becerra-Montejano, J. Y., & Camarena-Vázquez, M. R. (2025). *Control de mano robótica usando señales electromiográficas superficiales (EMG's)*. TecNM, Año 1(2), 18–31.
- Electromiografía - Mayo Clinic. (s. f.). <https://www.mayoclinic.org/es/tests-procedures/emg/about/pac-20393913>
- Guzmán-Muñoz, E., & Méndez-Rebolledo, G. (2018). Electromiografía en las Ciencias de la Rehabilitación. <https://www.redalyc.org/journal/817/81759607022/html/>
- Aparicio, M. V. (2005). Electromiografía cinesiológica. *Rehabilitación*, 39(6), 255-264. [https://doi.org/10.1016/s0048-7120\(05\)74359-0](https://doi.org/10.1016/s0048-7120(05)74359-0)
- Welcome to SENIAM. (s. f.). <http://www.seniam.org/>
- Doctordulitel. (2015, 8 junio). ¿De qué está formado el músculo? - Fisiomuro. Fisiomuro. https://www.fisiomuro.com/de-que-esta-formado-el-musculo/#%C2%BFDE_QUE_ESTA_FORMADO_UN_MUSCULO
- López, M. J. (2023, 6 julio). Placa neuromuscular: qué es y cómo funciona. Mejor Con Salud. <https://mejorconsalud.as.com/placa-neuromuscular/>

- Potencial de acción. (2023, 30 octubre). Kenhub. <https://www.kenhub.com/es/library/fisiologia/potencial-de-accion>
- Visible Body, part of Cengage Learning. (s. f.). Contracciones musculares | Aprenda anatomía muscular. <https://www.visiblebody.com/es/learn/muscular/muscle-contractions>
- Qué es el potencial de acción, potencial de membrana, gráfica de potencial de acción. (s. f.). <https://es.moleculardevices.com/applications/patch-clamp-electrophysiology/what-action-potential>
- Tabernig, C. B., Acevedo, R. C., & Fernández, J. M. (2007). INFLUENCIA DE LA FATIGA MUSCULAR EN LA SEÑAL ELECTROMIOGRÁFICA DE MÚSCULOS ESTIMULADOS ELÉCTRICAMENTE. *Revista EIA*, 7, 111–119. <https://www.researchgate.net/publication/237024758>
- Del Olmo, M., & Domingo, R. (2020). EMG Characterization and Processing in Production Engineering. *Materials*, 13(24), 5815. <https://doi.org/10.3390/ma13245815>
- Reaz, M. B. I., Hussain, M. S., & Mohd-Yasin, F. (2006). Techniques of EMG signal analysis: detection, processing, classification and applications. *Biological Procedures Online*, 8(1), 11-35. <https://doi.org/10.1251/bpo115>
- Shaw, Laxmi & Bhaga, Sangeeta. (2012). Online EMG Signal Analysis for diagnosis of Neuromuscular diseases by using PCA and PNN.. *International Journal Of Engineering Science and Technology* 0975-5462. 4. 4453-4459.
- Músculos del antebrazo. (2023, 23 noviembre). Kenhub. <https://www.kenhub.com/es/library/anatomia-es/musculos-del-antebrazo>

- Oiseth, S., Jones, L., & Maza Guia, E. (2025, 4 abril). Síndrome compartimental. Lecturio. <https://www.lecturio.com/es/concepts/sindrome-compartmental/>
- Nysora. (2022, 29 junio). Antebrazo. NYSORA. https://www.nysora.com/es/m%C3%BAsculo-esquel%C3%A9tico/antebrazo/#toc_5--WAD-M-VIL
- Body Anatomy: Upper extremity muscles | The Hand Society. (s. f.). <https://www.assh.org/handcareprod/safety/muscles#Wrist%20Forearm0Forearm>
- Movimiento articular. (s. f.). YouTube. <https://www.youtube.com/@MovimientoArticular>
- Flexor Pollicis Brevis. (2020). Physiopedia. https://www.physio-pedia.com/Flexor_Pollicis_Brevis#cite_note-4
- Themes, U. (2016, 11 junio). Anterior Forearm. Basicmedical Key. <https://basicmedicalkey.com/anterior-forearm/#f0010>
- 08. Antebrazo | Enfermería. (2024, 4 enero). Enfermería. <https://enfermeria.top/apuntes/anatomia/miembro-superior/antebrazo/>
- Camacho Sánchez, Pablo (2025). Diseño de un módulo de acondicionamiento para EEG. Trabajo de Fin de Grado. Ingeniería de la Salud, Universidad de Málaga.
- Learn about the Serial Peripheral Interface (SPI) · VectorNav. (s. f.). <https://www.vectornav.com/resources/inertial-navigation-primer/hardware/synccomm>
- Muceli, S., & Merletti, R. (2024). Tutorial. Frequency Analysis of the Surface EMG signal: Best Practices. *Journal of Electromyography and Kinesiology*, 79, 102937. <https://doi.org/10.1016/j.jelekin.2024.102937>

- Adquirir una señal analógica: ancho de banda, teorema de muestreo de Nyquist y aliasing. (2006, 31 agosto). NI. <https://www.ni.com/es/shop/data-acquisition/measurement-fundamentals/analog-fundamentals/acquiring-an-analog-signal--bandwidth--nyquist-sampling-theorem-.html>
- ADS1198 data sheet, product information and support | TI.com. (s. f.). <https://www.ti.com/product/ADS1198?qgpn=ads1198>
- Universitat Politècnica de València - UPV. (2015, 12 diciembre). Cuantificación | 12/84 | UPV [Video]. YouTube. <https://www.youtube.com/watch?v=sHhhSHpxvgE>
- Miyara, F., & Lahoz, L. A. (2003). INTRODUCCIÓN AL ANÁLISIS FRECUENCIAL Y AL RUIDO ELÉCTRICO. <https://www.fceia.unr.edu.ar/enica3/ruido-t>
- Transformación de Fourier: qué es. Diccionario CUN. (s. f.). <https://www.cun.es>. <https://www.cun.es/diccionario-medico/terminos/transformacion-fourier>
- Sörnmo, L., & Laguna, P. (2005). Chapter 7 – ECG signal processing. En L. Sörnmo & P. Laguna (Eds.), *Bioelectrical signal processing in cardiac and neurological applications* Academic Press. <https://www.sciencedirect.com/science/article/pii/B9780124375529500076>
- Oppenheim, A. V., Schafer, R. W., & Buck, J. R. (2010). *Filter design techniques* (Cap. 7, Sección 7.6 Comparisons on IIR and FIR discrete-time filters). En *Discrete-time signal processing*. (3.^a ed., pp. 439-511). Prentice Hall. https://research.iaun.ac.ir/pd/naghsh/pdfs/UploadFile_2230.pdf
- iirnotch - (To be removed) Second-order IIR notch filter - MATLAB. (s. f.). <https://es.mathworks.com/help/releases/R2025a/dsp/ref/iirnotch.html#d126e5018>

- Carrasco, R. (2024, 12 septiembre). Filtros de paso de banda. data-alliance. Recuperado 1 de septiembre de 2025, de <https://www.data-alliance.net/es/filtros-de-paso-de-banda>
- Análisis de señales fisiológicas: Técnicas | StudySmarter. (s. f.). StudySmarter ES. <https://www.studysmarter.es/resumenes/ingenieria/ingenieria-biomedica/analisis-de-senales-fisiologicas/>
- Butterworth Filter - an overview | ScienceDirect Topics. (s.f.). Www.sciencedirect.com. <https://www.sciencedirect.com/topics/engineering/butterworth-filter>
- Moctar, S. M. S., Rida, I., & Boudaoud, S. (2023, 1 junio). Time-domain features for sEMG signal classification: A brief survey. <https://hal.science/hal-04199535v1>
- Comparison Classification of EMG Signals in the Time Domain and Time-Frequency Domain. (2023, 10 octubre). IEEE Conference Publication | IEEE Xplore. <https://ieeexplore.ieee.org/document/10346962>
- Signal Features - MATLAB & Simulink. (2025). Mathworks.com. <https://es.mathworks.com/help/predmaint/ug/signal-features.html>
- Papandrea, P. (2024, 12 septiembre). *Advanced Analysis and Interpretation of the EMG Signal*. Cometa Systems. <https://www.cometasystems.com/advanced-analysis-and-interpretation-of-the-emg-signal/>
- Too, Jingwei & Abdullah, Abdul Rahim & Tengku Zawawi, Tengku Nor Shuhada & Mohd Saad, Norhashimah & Musa, Haslinda. (2017). Classification of EMG Signal Based on Time Domain and Frequency Domain Features. 1. 25-30. https://www.researchgate.net/publication/329011789_Classification_of_EMG_Signal_Based_on_Time_Domain_and_Frequency_Domain_Features#:~:text=The%20purpose%20of%20this%20paper,performance%20in%20discriminating%20EMG%20signal

- signalTimeFeatureExtractor - Streamline signal time feature extraction - MATLAB. (s. f.).
<https://mathworks.com/help/releases/R2025a/signal/ref/signaltimefeatureextractor.html>
- signalFrequencyFeatureExtractor - Streamline signal frequency feature extraction - MATLAB. (s. f.).
<https://es.mathworks.com/help/releases/R2025a/signal/ref/signalfrequencyfeatureextractor.html>
- Manu Duque. (2025, 9 marzo). *Normalización puntuación estándar* - Manu Duque. Manu Duque ▶ Blog Social Media & SEO. <https://www.manuduque.com/enciclopedia/normalizacion-puntuacion-estandar/>
- Classification Learner - Train models to classify data using supervised machine learning - MATLAB. (s. f.).
<https://es.mathworks.com/help/releases/R2025a/stats/classificationlearner-app.html>
- Sanahuja, P. M. (s. f.). *Entendiendo la curva ROC y el AUC: Dos medidas del rendimiento de un clasificador binario que van de la mano.* – Pol Martí Sanahuja. <https://polmartisanahuja.com/entendiendo-la-curva-roc-y-el-auc-dos-medidas-del-rendimiento-de-un-clasificador-binario-que-van-de-la-mano/>
- *Modelos de Ensamble (Ensemble Models) | AI Planet (formerly DPhi).* (s. f.). AI Planet (Formerly DPhi). <https://aiplanet.com/learn/supervised-learning-regression-es/bosques-aleatorios-random-forest/1736/modelos-de-ensamble-ensemble-models>
- Ashour, A. S., Guo, Y., Hawas, A. R., & Xu, G. (2018). Ensemble of subspace discriminant classifiers for schistosomal liver fibrosis staging in mice microscopic images. *Health Information Science And Systems*, 6(1). <https://doi.org/10.1007/s13755-018-0059-8>

- Ibm. (2025, 18 febrero). Análisis discriminante lineal. IBM. <https://www.ibm.com/es-es/think/topics/linear-discriminant-analysis>
- Arce, J. I. B. (2024, 13 septiembre). La matriz de confusión y sus métricas. Juan Barrios. <https://www.juanbarrios.com/la-matriz-de-confusion-y-sus-metricas/>
- Visualize and Assess Classifier Performance in Classification Learner - MATLAB & Simulink. (s. f.). <https://es.mathworks.com/help/releases/R2025a/stats/assess-classifier-performance.html#bun1aps-1>
- Tudela Roldán, Alberto José (2015). Montaje y control de un brazo robótico impreso en 3D. Proyecto de Fin de Carrera. Ingeniería de Telecomunicaciones, Universidad de Málaga.

Apéndice A: Código fuente

Procesamiento de los datos de entrenamiento

1. Configuración inicial

```
rootFolder = 'C:\Users\raque\Documents\Uni\Importante\TFG\Señales (1)\Señales';

multiplicationFactor = 0.0061;

signals = struct();

movementFolders = dir(rootFolder);
movementFolders = movementFolders([movementFolders.isdir] & ~ismember({movementFolders.name},
{'.' , '..'}));

for i = 1:length(movementFolders)
    currentMovementName = movementFolders(i).name;
    currentMovementPath = fullfile(rootFolder, currentMovementName);
    validMovementFieldName = matlab.lang.makeValidName(currentMovementName);

    signals.(validMovementFieldName) = struct();

    txtFiles = dir(fullfile(currentMovementPath, '*.txt'));

    for k = 1:length(txtFiles)
        currentFile = fullfile(txtFiles(k).folder, txtFiles(k).name);
        [~, nameOnly, ~] = fileparts(currentFile);
        validFileName = matlab.lang.makeValidName(nameOnly);
        fid = fopen(currentFile, 'r');
        rawData = textscan(fid, repmat('%f', 1, 8), 'Delimiter', ' ', 'CollectOutput', true);
        fclose(fid);
        rawData = rawData{1}; % Convertir de celda a matriz Nx8

        if size(rawData, 2) < 8
            fprintf("⚠ Archivo '%s' tiene solo %d columnas. Se esperaban 8.\n", ...
                currentFile, size(rawData, 2));
        end

        processedData = rawData * multiplicationFactor;

        signals.(validMovementFieldName).(validFileName) = processedData;
    end
end
```

2. Centrado en cero para eliminar la componente continua

```
filtradoDC = struct();

% Obtener los nombres de los movimientos
movementNames = fieldnames(signals);

% Itera sobre cada movimiento
for m_idx = 1:length(movementNames)
    currentMovementName = movementNames{m_idx};
    currentMovementStruct = signals.(currentMovementName);

    % Inicializa el sub-struct para este movimiento en 'filtradoDC'
    filtradoDC.(currentMovementName) = struct();

    % Obtener los nombres de los archivos dentro de este movimiento
    fileNamesInMovement = fieldnames(currentMovementStruct);

    % Itera sobre cada archivo
    for f_idx = 1:length(fileNamesInMovement)
        currentFileName = fileNamesInMovement{f_idx};
        rawData = currentMovementStruct.(currentFileName); % Accede a los datos del archivo

        % Comprobar si alguna columna contiene NaN
        colswithNaN = any(isnan(rawData), 1); % vector lógico por columnas
        if any(colswithNaN)
            fprintf("⚠ Archivo '%s/%s' contiene NaN en las columnas: %s\n", ...
                currentMovementName, currentFileName, mat2str(find(colswithNaN)));
        end

        % Calcular la media por canal para
        mean_per_channel = mean(rawData, 1);

        % Restar la media a cada canal
        processedData = rawData - mean_per_channel;

        % Guardar los datos procesados en el nuevo struct
        filtradoDC.(currentMovementName).(currentFileName) = processedData;
    end
end
```

3. Transformada Rápida de Fourier

```
fs=2000;
s=filtradoDC.Reposo.reposo_u1_rep5(:,1);
x=s(:,1);

L=length(x);
Y=fft(x);
P2=abs(Y/L);
P1=P2(1:L/2+1);

P1(2:end-1)=2*P1(2:end-1);
f=fs*(0:(L/2))/L;
```

```

figure()
plot(f,P1)

xlabel('Frecuencia (Hz)');
ylabel('Magnitud (dB)');
title('Transformada Rápida de Fourier (FFT)')
xlim([0 500]);
ylim([0 0.035]);

```

4. Filtro Notch

```

% Parámetros del Filtro Notch
Fs = 2000;           % Frecuencia de muestreo
F0 = 50;            % Frecuencia a eliminar
Q = 30;             % Factor de calidad
numPasses = 1;      % Número de veces que se aplica el filtro

% Cálculo de Coeficientes del Filtro
wo = F0 / (Fs/2);
bw = wo / Q;
[b, a] = iirnotch(wo, bw);

% Inicializa el struct de salida
filtrado_Notch = struct();

% Obtener nombres de los movimientos
movementNames = fieldnames(filtradoDC);
% Iterar sobre cada movimiento
for m_idx = 1:length(movementNames)
    currentMovementName = movementNames{m_idx};
    currentMovementStruct = filtradoDC.(currentMovementName);
    filtrado_Notch.(currentMovementName) = struct(); % Inicializa el sub-struct

    % Obtener nombres de los archivos dentro del movimiento
    fileNamesInMovement = fieldnames(currentMovementStruct);

    % Iterar sobre cada archivo
    for f_idx = 1:length(fileNamesInMovement)
        currentFileName = fileNamesInMovement{f_idx};
        originalSignal = currentMovementStruct.(currentFileName);
        filteredSignal = zeros(size(originalSignal));

        % Aplicar filtro a cada canal
        for ch = 1:size(originalSignal, 2)
            temp_channel_data = originalSignal(:, ch);
            for k = 1:numPasses
                temp_channel_data = filter(b, a, temp_channel_data);
            end
            filteredSignal(:, ch) = temp_channel_data;
        end
        filtrado_Notch.(currentMovementName).(currentFileName) = filteredSignal;
    end
end
end

```

5. Filtro paso banda

```
fs = 2000;          % Frecuencia de muestreo de 2000 Hz
fci = 50;           % Frecuencia de corte inferior (Hz)
fcs = 150;          % Frecuencia de corte superior (Hz)
orden = 5;          % Orden del filtro

% Filtro Butterworth pasa banda de 5º orden
wn = [fci fcs] / (fs / 2); % Normalizar
[b1, a1] = butter(orden, wn, 'bandpass');

paso_banda_Notch=struct();

% Obtener nombres de los movimientos
movementNames = fieldnames(filtrado_Notch);

% Iterar sobre cada movimiento
for m_idx = 1:length(movementNames)
    currentMovementName = movementNames{m_idx};
    currentMovementStruct = filtrado_Notch.(currentMovementName);

    paso_banda_Notch.(currentMovementName) = struct(); % Inicializa el sub-struct

    % Obtener nombres de los archivos dentro del movimiento
    fileNamesInMovement = fieldnames(currentMovementStruct);

    % Iterar sobre cada archivo
    for f_idx = 1:length(fileNamesInMovement)
        currentFileName = fileNamesInMovement{f_idx};
        originalSignal = currentMovementStruct.(currentFileName);

        filteredSignal = zeros(size(originalSignal));

        % Aplicar filtro a cada canal
        for ch = 1:size(originalSignal, 2)
            filteredSignal(:, ch) = filter(b1,a1,originalSignal(:,ch));
        end

        paso_banda_Notch.(currentMovementName).(currentFileName) = filteredSignal;
    end
end
```

6. Eliminación de subcarpetas del struct para facilitar el procesamiento

```
signalsNamed = struct();

movimientos = {'Abrir', 'Cerrar', 'Pronacion', 'Supinacion', 'Reposo', 'Pulgar', 'Dos_dedos'};

for i = 1:numel(movimientos)
    movName = movimientos{i};
    subStruct = paso_banda_Notch.(movName);
    signalNames = fieldnames(subStruct);

    for j = 1:numel(signalNames)
```

```

    sigName = signalNames{j};
    signal = subStruct.(sigName);

    % Crear campo con nombre 'movName_sigName' para que sea único
    fieldName = [sigName];
    signalsNamed.(fieldName) = signal;
end
end

reps_por_mov = 40; % repeticiones por movimiento

labels = cell(1, length(signalsNamed));

for i = 1:length(movimientos)
    start_idx = (i-1)*reps_por_mov + 1;
    end_idx = i*reps_por_mov;
    labels(start_idx:end_idx) = repmat(movimientos(i), 1, reps_por_mov);
end

```

7. Extracción de características

```

extractor_tiempo = signalTimeFeatureExtractor( ...
    "Mean", true, ...
    "RMS", true, ...
    "StandardDeviation", true, ...
    "ShapeFactor", true, ...
    "SNR", true, ...
    "THD", true, ...
    "SINAD", true, ...
    "PeakValue", true, ...
    "CrestFactor", true, ...
    "ClearanceFactor", true, ...
    "ImpulseFactor", true);

extractor_freq = signalFrequencyFeatureExtractor( ...
    "MeanFrequency", true, ...
    "MedianFrequency", true, ...
    "BandPower", true, ...
    "OccupiedBandwidth", true, ...
    "PowerBandwidth", true, ...
    "PeakAmplitude", true, ...
    "PeakLocation", true);

timeNames = { ...
    'Mean', 'RMS', 'StandardDeviation', 'ShapeFactor', ...
    'SNR', 'THD', 'SINAD', 'PeakValue', ...
    'CrestFactor', 'ClearanceFactor', 'ImpulseFactor'};

freqNames = { ...
    'MeanFrequency', 'MedianFrequency', 'BandPower', ...
    'OccupiedBandwidth', 'PowerBandwidth', ...
    'PeakAmplitude', 'PeakLocation'};

allFeatureNames = [timeNames, freqNames];

```

```

nChannels = 8;
columnNames = {};
for ch = 1:nChannels
    for f = 1:numel(allFeatureNames)
        columnNames{end+1} = sprintf('channel%d_%s', ch, allFeatureNames{f});
    end
end

fieldNames = fieldnames(signalsNamed);
nFields = numel(fieldNames);
resultMatrix = zeros(nFields, nChannels * numel(allFeatureNames)); % prealocar

for i = 1:nFields
    rep = signalsNamed.(fieldNames{i}); % matriz Nx8
    features_row = [];

    for ch = 1:nChannels
        signal = rep(:, ch);

        % verificar si hay valores no finitos
        invalidIdx = find(~isfinite(signal));
        if ~isempty(invalidIdx)
            fprintf("\u2609 Campo '%s', canal %d contiene valores no finitos en las líneas:
%s\n", ...
                fieldNames{i}, ch, mat2str(invalidIdx));
        end

        time_feats = extract(extractor_tiempo, signal);
        freq_feats = extract(extractor_freq, signal);

        channel_feats = [time_feats, freq_feats];
        features_row = [features_row, channel_feats];
    end

    resultMatrix(i, :) = features_row;
end

resultsTable = array2table(resultMatrix, ...
    'VariableNames', columnNames, ...
    'RowNames', fieldNames);

```

8. Adición de la columna de clase

```

n = height(resultsTable);
bloque = 40;
numBloques = n / bloque;

```

```

movimientos = repmat({'Abrir', 'Cerrar', 'Pronacion', 'Supinacion', 'Reposo', 'Pulgar',
'Dos_dedos'}, 1, ceil(numBloques/2));
movimientos = movimientos(1:numBloques); % asegurarse que tenga exactamente numBloques

columna_mov = repelem(movimientos, bloque)';

resultsTable.Movimiento = columna_mov;

```

9. Estandarización de características

```

X = resultsTable(:,1:end-1);
X_mat = table2array(X); % convertir a matriz

% Normalizar
[Xz, mu, sigma] = zscore(X_mat);

rowNames=resultsTable.Properties.RowNames;

standarized_features= array2table(Xz, 'VariableNames', X.Properties.VariableNames, 'RowNames',
rowNames);

cat_name = resultsTable.Properties.VariableNames{end};

Cat = resultsTable.(cat_name);

standarized_features.(cat_name) = Cat;

save('C:\Users\raque\Documents\Uni\Importante\TFG\Señales
(1)\Señales\Pruebas\stand_params.mat', 'mu', 'sigma')

```

Script principal: adquisición, procesamiento, clasificación y control del brazo

```

% Abrir puerto para el brazo robótico
if ~exist('puerto_robot', 'var') || isempty(puerto_robot)
    puerto_robot = serialport("COM4",115200);
    configureTerminator(puerto_robot, "CR");
end
%Adquirir señal
leer_serie("COM3", 115200, 'pruebas.txt')

% Procesar señal
features = preprocesamiento(signal);

% Clasificar gesto
clase = clasificador(features);
% Mover el robot
movimiento_robot(clase, puerto_robot);

```

Función de lectura de señales

```
function datos = leer_serie(puerto, baudrate, filename)
    % Configurar puerto serie
    s = serialport(puerto, baudrate);
    configureTerminator(s,"LF");
    flush(s);

    % Buffers
    lineas = strings(0);
    datos = [];

    disp("Leyendo datos... pulsa una tecla en la consola para detener.");

    % Bucle de lectura
    while ~kbhit()
        try
            linea = readline(s); % lee la línea como string
            lineas(end+1,1) = linea;

            % convertir a números
            nums = str2double(split(strtrim(linea)));
            if numel(nums) == 8
                datos = [datos; nums'];
            end
        catch
            % ignora errores puntuales
        end
    end

    % Guardar archivo de texto
    writelines(lineas, filename);

    % Convertir a double por seguridad
    datos = double(datos);

    % Liberar puerto
    clear s

    disp("Lectura detenida. Datos guardados en " + filename);
end

function key = kbhit()
    % Detectar pulsación de tecla en consola MATLAB
    key = false;
    pause(0.01);
    w = waitforbuttonpress;
    if w == 1
        key = true;
    end
end
```

Función de procesamiento

```
function std_features=preprocesamiento(signal)
%Conversión de señal a mV
signal_mv=signal*0.0061;

%Filtrado DC--> centrado de la señal
%Cálculo de la media
[filas, canales] = size(signal_mv);
media = zeros(1, canales);

for i = 1:canales
    canal = signal_mv(:, i);           % Extraer vector columna
    media(i) = mean(signal_mv(:,i));   % Calcular características
end

%Eliminación de ruido DC
filtradoDC = zeros(size(signal_mv));

for i=1:filas
    for j = 1:canales
        filtradoDC(i, j) = signal_mv(i, j) - media(j);
    end
end

%Filtro notch
Fs = 2000;           % Frecuencia de muestreo (Hz)
F0 = 50;            % Frecuencia a eliminar (Hz)
Q = 30;             % Factor de calidad

% Diseño del filtro Notch
wo = F0 / (Fs/2);   % Frecuencia normalizada
bw = wo / Q;        % Ancho de banda normalizado
[b, a] = iirnotch(wo, bw); % Coeficientes del filtro

% Inicializar matriz de salida
filtrado_Notch = zeros(size(filtradoDC));

% Aplicar filtro a cada canal
for ch = 1:size(filtradoDC, 2)
    temp = filtradoDC(:, ch);
    temp = filter(b, a, temp);
    filtrado_Notch(:, ch) = temp;
end

%Filtro paso banda + Notch
% Parámetros del filtro paso banda
fs = 2000;           % Frecuencia de muestreo (Hz)
fci = 50;            % Frecuencia de corte inferior (Hz)
fcs = 150;           % Frecuencia de corte superior (Hz)
orden = 5;           % Orden del filtro

% Diseño del filtro Butterworth pasa banda
wn = [fci fcs] / (fs / 2); % Frecuencias normalizadas
```

```

[b1, a1] = butter(orden, wn, 'bandpass'); % Coeficientes del filtro

% Inicializar matrices de salida
paso_banda = zeros(size(filtradoDC));
paso_banda_Notch = zeros(size(filtrado_Notch));

% Aplicar el filtro a cada canal de la matriz filtradoDC
for ch = 1:size(filtradoDC, 2)
    paso_banda(:, ch) = filter(b1, a1, filtradoDC(:, ch));
end

% Aplicar el filtro a cada canal de la matriz filtrado_Notch
for ch = 1:size(filtrado_Notch, 2)
    paso_banda_Notch(:, ch) = filter(b1, a1, filtrado_Notch(:, ch));
end

```

```

%Extracción de características
% === Paso 1: Crear extractores ===
extractor_tiempo = signalTimeFeatureExtractor( ...
    "Mean", true, ...
    "StandardDeviation", true, ...
    "RMS", true, ...
    "ShapeFactor", true, ...
    "CrestFactor", true, ...
    "PeakValue", true, ...
    "ImpulseFactor", true, ...
    "ClearanceFactor", true, ...
    "SNR", true, ...
    "SINAD", true, ...
    "THD", true);

extractor_freq = signalFrequencyFeatureExtractor( ...
    "MeanFrequency", true, ...
    "MedianFrequency", true, ...
    "BandPower", true, ...
    "OccupiedBandwidth", true, ...
    "PowerBandwidth", true, ...
    "PeakAmplitude", true, ...
    "PeakLocation", true);

% === Paso 2: Obtener nombres de características ===
nombres_tiempo = { ...
    'Mean', ...
    'RMS', ...
    'StandardDeviation', ...
    'ShapeFactor', ...
    'SNR', ...
    'THD', ...
    'SINAD', ...
    'PeakValue', ...
    'CrestFactor', ...
    'ClearanceFactor', ...
    'ImpulseFactor', ...
};

nombres_freq = { ...

```

```

'MeanFrequency', ...
'MedianFrequency', ...
'BandPower', ...
'OccupiedBandwidth', ...
'PowerBandwidth', ...
'PeakAmplitude', ...
'PeakLocation'];
nombres_total = [nombres_tiempo, nombres_freq];

% === Paso 3: Inicializar matriz de características ===
canales = size(paso_banda_Notch, 2);
num_feat_total = numel(nombres_total);
matriz_caracteristicas = zeros(canales, num_feat_total);

% === Paso 4: Extraer características por canal ===
for i = 1:canales
    canal = paso_banda_Notch(:, i);
    TFE = extract(extractor_tiempo, canal);
    FFE = extract(extractor_freq, canal);
    matriz_caracteristicas(i, :) = [TFE, FFE];
end

% === Paso 5: Generar nombres tipo "ChannelXCaracteristica" ===
nuevo_nombre_columnas = {};
for canal = 1:canales
    for k = 1:numel(nombres_total)
        nuevo_nombre = ['channel' num2str(canal) '_' nombres_total{k}];
        nuevo_nombre_columnas{end+1} = nuevo_nombre;
    end
end

% Convertir matriz en fila única (aplanada por canal y característica)
tabla_final = array2table(reshape(matriz_caracteristicas', 1, []), ...
    'VariableNames', nuevo_nombre_columnas);
features=tabla_final;

% === Orden correcto de las características (por canal) ===
orden_deseado = { ...
    'Mean', 'RMS', 'StandardDeviation', 'ShapeFactor', ...
    'SNR', 'THD', 'SINAD', 'PeakValue', ...
    'CrestFactor', 'ClearanceFactor', 'ImpulseFactor', ...
    'MeanFrequency', ...
    'MedianFrequency', ...
    'BandPower', ...
    'OccupiedBandwidth', ...
    'PowerBandwidth', ...
    'PeakAmplitude', ...
    'PeakLocation'};

% === Generar nuevo orden de columnas ===
nuevo_orden_columnas = {};
for canal = 1:canales
    for k = 1:length(orden_deseado)
        nombre = ['channel' num2str(canal) '_' orden_deseado{k}];
        nuevo_orden_columnas{end+1} = nombre;
    end
end

```

```

% === Reordenar la tabla ===
features = features(:, nuevo_orden_columnas);

%Estandarización de las características
% Cargar mu y sigma previamente guardados
load('C:\Users\raque\Documents\Uni\Importante\TFG\Señales
(1)\Señales\Pruebas\stand_params.mat', 'mu', 'sigma');

% Cargar nuevos datos
% Supongamos que tienes una tabla sin la columna categórica
Xnew = features; % Si ya está en formato tabla

% Convertir a matriz numérica
Xnew_mat = table2array(Xnew);

% Aplicar estandarización manualmente
if length(mu) ~= width(features)
    error('La longitud de mu no coincide con el número de columnas de features');
end
Xnew_z = (Xnew_mat - mu) ./ sigma;

% volver a tabla con los mismos nombres de columna
std_features = array2table(Xnew_z, 'VariableNames', Xnew.Properties.VariableNames);

end

```

Función de clasificación de gestos

```

function clase=clasificador(std_features)
    load("Ensemble_Subspace_Discriminant.mat")
    clase =Ensemble_Subspace_Discriminant.predictFcn(std_features);
    clase = string(mode(categorical(clase)));
end

```

Función de control del robot

```

function movimiento_robot(clase,puerto)

    persistent primeraVez
    if isempty(primeravez)
        writeline(puerto, "p");
        primeraVez = false;
    end
    switch clase
        case "abrir"
            writeline(puerto, "O");
            disp("Abriendo la mano")
        case "cerrar"
            writeline(puerto, "C");
            disp("Cerrando la mano")
    end
end

```

```
case "dos_Dedos"  
    writeline(puerto, "F,180,180,0,0,180");  
    disp("Extendiendo índice y corazón")  
case "pronación"  
    writeline(puerto, "W,0");  
    disp("Pronando muñeca")  
case "supinación"  
    writeline(puerto, "W,180");  
    disp("Supinando muñeca")  
case "pulgar"  
    writeline(puerto, "F,180,180,180,180,0");  
    disp("Extendiendo pulgar")  
case "reposo"  
    writeline(puerto, "F,90,90,90,90,90");  
    writeline(puerto, "W,90");  
    disp("Reposo")  
end  
end
```




UNIVERSIDAD
DE MÁLAGA

| uma.es

E.T.S de Ingeniería Informática
Bulevar Louis Pasteur, 35
Campus de Teatinos
29071 Málaga

E.T.S. DE INGENIERÍA INFORMÁTICA