

Universidad Autónoma de Querétaro

Facultad de Ingeniería

Ingeniería Biomédica

**Dispositivo de asistencia para el control de dirección de silla de ruedas por medio de
señales EMG**

TESIS

Que como parte de los requisitos para obtener el grado de

Ingeniero Biomédico

Presenta:

Natalia Margarita Rendón Caballero

Dirigido por:

Dr. Marcos Romo Avilés

SINODALES

Dr. Marcos Romo Avilés

Presidente

Dr. José Manuel Álvarez Alvarado

Secretario

Dr. Juvenal Rodríguez Reséndiz

Vocal

Dra. Gloria Nélide AVECILLA RAMÍREZ

Suplente

Dra. Luz María Sánchez Reyes

Suplente

Campus Aeropuerto
Querétaro, QRO, México.

Agosto 2025

La presente obra está bajo la licencia:
<https://creativecommons.org/licenses/by-nc-nd/4.0/deed.es>



CC BY-NC-ND 4.0 DEED

Atribución-NoComercial-SinDerivadas 4.0 Internacional

Usted es libre de:

Compartir — copiar y redistribuir el material en cualquier medio o formato

La licenciante no puede revocar estas libertades en tanto usted siga los términos de la licencia

Bajo los siguientes términos:



Atribución — Usted debe dar [crédito de manera adecuada](#), brindar un enlace a la licencia, e [indicar si se han realizado cambios](#). Puede hacerlo en cualquier forma razonable, pero no de forma tal que sugiera que usted o su uso tienen el apoyo de la licenciante.



NoComercial — Usted no puede hacer uso del material con [propósitos comerciales](#).



SinDerivadas — Si [remezcla, transforma o crea a partir](#) del material, no podrá distribuir el material modificado.

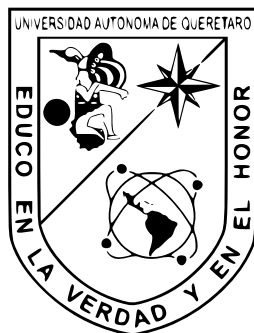
No hay restricciones adicionales — No puede aplicar términos legales ni [medidas tecnológicas](#) que restrinjan legalmente a otras a hacer cualquier uso permitido por la licencia.

Avisos:

No tiene que cumplir con la licencia para elementos del material en el dominio público o cuando su uso esté permitido por una [excepción o limitación](#) aplicable.

No se dan garantías. La licencia podría no darle todos los permisos que necesita para el uso que tenga previsto. Por ejemplo, otros derechos como [publicidad, privacidad, o derechos morales](#) pueden limitar la forma en que utilice el material.

UNIVERSIDAD AUTÓNOMA DE QUERÉTARO



FACULTAD DE INGENIERÍA
INGENIERA BIOMÉDICA

Dispositivo de asistencia para el control de dirección de silla de ruedas por medio de señales EMG

TESIS

Que como parte de los requisitos para obtener el grado de
Ingeniero Biomédico

Presenta:

NATALIA MARGARITA RENDÓN CABALLERO

Dirigido por:

DR. MARCOS ROMO AVILÉS

Co-Director:

DR. JOSÉ MANUEL ÁLVAREZ ALVARADO

Vocal:

DR. JUVENAL RODRÍGUEZ RESÉNDIZ

Suplente 1:

DRA. GLORIA NÉLIDA AVECILLA RAMÍREZ

Suplente 2:

DRA. LUZ MARÍA SÁNCHEZ REYES

Querétaro, Qro. a Agosto de 2025

A cada persona que sembró una semilla de inspiración en este camino, que me brindó su consejo en los momentos de duda y celebró conmigo cada pequeño triunfo. Este logro es un reflejo de su apoyo incondicional, su paciencia infinita y su esfuerzo conjunto. A ustedes, que me impulsaron a seguir adelante incluso cuando la meta parecía lejana, les dedico este trabajo.

Agradecimientos

A la Facultad de Ingeniería, por todos los conocimientos y oportunidades brindadas a lo largo de mi carrera. Al Fondo de Proyectos Especiales de Rectoría (FOPER) 2024, cuyo apoyo ha sido clave en la realización de este trabajo.

A mi director de tesis, el Dr. Marcos Romo Avilés, por su invaluable orientación y constante guía, que han sido fundamentales para este trabajo culminado. Gracias por compartir su pasión por la investigación, por su dedicación y por inspirarme a alcanzar este resultado del que me siento tan orgullosa.

Agradezco a mis asesores, el Dr. Manuel y el Dr. Juvenal, por su paciencia, los conocimientos brindados y la confianza al darme la oportunidad de trabajar en conjunto para la creación de esta tesis.

A mis padres, Isidro y Margarita, cuyo amor infinito, sacrificio y dedicación me han enseñado el valor del trabajo, el esfuerzo y la perseverancia. Por darme la libertad de crecer y forjar mi propio camino. Todo lo que soy es reflejo de ustedes.

A mis hermanas, Nereyda y Aurea, porque con ustedes la vida siempre es más ligera y divertida; por ser mi ejemplo y motivación, por compartir su sabiduría en la vida y el trabajo, y por recordarme con su cariño que soy capaz de lograr lo que me proponga.

A mi primo Antonio, o más bien, mi hermano. Gracias por compartir conmigo no solo un hogar, sino también risas, sueños y momentos difíciles. Por estar siempre ahí, por tu apoyo incondicional y por esos abrazos que llegaron justo cuando más los necesitaba.

A mis amigos que se convirtieron en familia mientras estuve lejos de casa, Maritza, Michelle, Ana y Portos, gracias por todas las aventuras vividas, los momentos de enseñanza, estrés y aprendizaje, acompañarme en cada paso y por demostrarme que, al final todo salió.

A Karla, mi constante más valiosa, que a pesar de la distancia, me llenó de cariño y apoyo inquebrantable.

A Alejandro, quien he elegido como compañero de vida, gracias por ser mi refugio y mi motivación. Por siempre creer en que lograría todo lo que hoy celebramos juntos. *Solo te dejaría por Charles Leclerc.*

Resumen

Existe un gran número de personas con discapacidad motriz que requieren de una silla de ruedas, sin embargo, el uso de estas implica un mayor desgaste físico a las extremidades superiores del usuario. Mediante el procesamiento, así como el reconocimiento y clasificación de patrones en señales electromiográficas (EMG), es posible reducir este desgaste físico a sólo determinados movimientos de la mano.

Por ello, el presente proyecto incorpora la identificación de señales EMG adquiridas de la mano derecha a partir de un clasificador con señales predichas de autoencoders.

Un circuito de instrumentación compuesto de 3 sensores se diseñó para la adquisición de señales EMG a partir de cinco movimientos de la mano. Posteriormente, se creó una base de datos con 20 sujetos sanos (sin afecciones cardiovasculares, enfermedades del aparato locomotor ni del sistema nervioso, sin amputaciones ni con obesidad) entre 18 y 40 años. Estas señales se segmentaron en 200 ms y ventanearon digitalmente con 75 % de traslape. La base de datos se dividió en 72 % entrenamiento, 8 % validación y 20 % test. El conjunto de validación se utilizó para extraer características a partir de tres autoencoders (uno para cada sensor), donde la señal fue codificada y decodificada con una pérdida mínima de hasta 3.1×10^{-4} . A partir de este proceso, se extrajo el vector latente, con el cual se predijeron las señales del conjunto de entrenamiento. Con estas, se entrenó una red neuronal densa para la clasificación de los cinco movimientos.

Se obtuvo una eficiencia del 93.84 % en la red neuronal, calculada a partir de las métricas de sensibilidad, precisión, especificidad y eficiencia para cada clase.

Abstract

There are a large number of people with motor disabilities who require a wheelchair, however, the use of these involves a greater physical wear to the user's upper limbs. By processing, as well as recognizing and classifying patterns in electromyographic (EMG) signals, it is possible to reduce this physical wear to only certain hand movements.

Therefore, the present project incorporates the identification of EMG signals acquired from the right hand from a classifier with predicted signals from autoencoders.

An instrumentation circuit composed of 3 sensors was designed for the acquisition of EMG signals from five hand movements. Subsequently, a database of 20 healthy subjects (without cardiovascular conditions, locomotor or nervous system diseases, amputations or obesity) between 18 and 40 years of age was created. These signals were segmented into 200 ms and digitally windowed with 75 % overlap. The database was divided into 72 training, 8 validation and 20 test sets. The validation set was used to extract features from three autoencoders (one for each sensor), where the signal was encoded and decoded with a minimum loss of up to 3.1×10^{-4} . From this process, the latent vector was extracted, with which the signals of the training set were predicted. With these, a dense neural network was trained for the classification of the five movements.

An efficiency of 93.84 % was obtained in the neural network, calculated from the metrics of sensitivity, precision, specificity and efficiency for each class.

Índice general

1. Introducción	1
1.1. Antecedentes	2
1.2. Justificación	6
1.3. Descripción del problema	7
1.4. Hipótesis	8
1.5. Objetivo General	8
1.6. Objetivos Específicos	8
2. Fundamentación teórica	9
2.1. Contracción Muscular	9
2.2. Adquisición de señal EMG	9
2.3. Señal EMG	10
2.4. Amplificación	11
2.5. Filtrado	11
2.6. Segmentación	11
2.7. Inteligencia Artificial	13
2.7.1. Aprendizaje Automático	13
2.7.2. Aprendizaje Profundo	13
2.7.3. Autoencoder	20
2.8. Validación del modelo	21
3. Metodología	23
3.1. Sistema de adquisición	24
3.1.1. Diseño del circuito de alimentación	24
3.1.2. Diseño del circuito de adquisición de señales EMG	24
3.1.3. Filtrado	24
3.1.4. Eliminación de interferencias	26
3.1.5. Salida para digitalización	26
3.1.6. Manufactura de PCB	27
3.1.7. Digitalización de la señal	27
3.1.8. Validación e integración del sistema	27
3.2. Base de datos	28
3.3. Segmentación de las señales	32
3.4. Diseño del sistema de clasificación	32

3.4.1. Codificador y Decodificador	33
3.4.2. Clasificador EMG	33
3.5. Interfaz Gráfica	34
3.6. Validación del modelo	35
4. Resultados	36
4.1. Sistema de adquisición	36
4.2. Segmentación y ventaneo	37
4.3. Codificador y decodificador	41
4.4. Clasificador EMG	44
4.5. Interfaz gráfica	50
4.6. Discusión	51
5. Conclusiones	54
5.1. Trabajos Futuros	54
Referencias	62

Índice de figuras

1.1. Red bibliométrica.	4
2.1. Conexión de la neurona a la fibra muscular por medio del axón.	10
2.2. Técnicas de ventaneo.	12
2.3. Esquema de una neurona artificial.	14
2.4. Esquema de una red neuronal y sus parámetros.	15
2.5. Arquitectura LSTM.	19
2.6. Arquitectura GRU.	20
3.1. Esquema de la metodología general para el clasificador de señales EMG.	23
3.2. Operador de bionstrumentación.	24
3.3. Filtro Pasa altas.	25
3.4. Filtro Pasa bajas.	26
3.5. Filtro Notch.	26
3.6. Offset y salida de la señal.	27
3.7. Sistema de adquisición de señales EMG compuesto por dos baterías de litio, una placa de alimentación, 3 placas EMG y el DAQ.	28
3.8. Posición de inicio que adoptaron los sujetos durante la toma de señales.	29
3.9. Colocación de electrodos en el músculo a) Flexor común de los dedos, b) extensor común de los dedos, y c) flexor del pulgar.	30
3.10. Protocolo de adquisición.	31
3.11. Toma de señales para la base de datos.	31
4.1. Manufactura de la PCB de alimentación.	36
4.2. Sistema de adquisición EMG	37
4.3. Segmentación de las señales de los diferentes movimientos de la mano.	38
4.4. Comparación de la ventana 1 y la ventana 2 para los 5 movimientos.	40
4.5. Predicción y señal real del sensor 1 del autoencoder global.	42
4.6. Predicción y señal real del sensor 2 del autoencoder global.	42
4.7. Predicción y señal real del sensor 3 del autoencoder global.	43
4.8. Predicción y señal real del sensor 1 del autoencoder individual.	43
4.9. Predicción y señal real del sensor 2 del autoencoder individual.	43
4.10. Predicción y señal real del sensor 3 del autoencoder individual.	44
4.11. Representación del modelo de clasificación.	46
4.12. Gráfica de la precisión en entrenamiento y validación.	47

4.13. Gráfica de la pérdida en entrenamiento y validación	48
4.14. Matriz de confusión para las 5 clases.	49
4.15. Interfaz gráfica.	51

Índice de tablas

1.1. Resumen de modelos de ANN en bibliografía consultada.	3
2.1. Funciones de activación obtenidas	15
2.2. Definición de variables empleadas en modelos de redes neuronales.	17
2.3. Definición de subíndices empleadas en modelos de redes neuronales	17
2.4. Definición de variables empleadas en la validación del modelo	22
3.1. Músculos sensados, su ubicación y función	30
4.1. Resumen de la arquitectura y parámetros del clasificador	45
4.2. Métricas de evaluación del modelo por clase	50
4.3. Comparación de los trabajos previos con el actual.	52

Introducción

Aproximadamente tres millones de personas tienen dificultad para caminar, subir y bajar escaleras [1]. Por ello las personas que no pueden caminar, necesitan una silla de ruedas para moverse. Sin embargo, las sillas de ruedas convencionales que ofrece el mercado requieren de un gran esfuerzo físico por parte de los usuarios [2].

Por lo anterior, la adquisición de señales musculares en tiempo real y la implementación de redes neuronales recurrentes emergen como una solución prometedora para el desarrollo de un sistema de control de dirección en sillas de ruedas. La técnica de electromiografía permite registrar la actividad eléctrica de la contracción muscular y obtener señales EMG [3], las cuales pueden ser procesadas mediante redes neuronales recurrentes para identificar patrones complejos en los datos [4]. Durante su fase de entrenamiento, estos modelos requieren la definición de hiperparámetros, los cuales determinan su arquitectura y comportamiento [5]. Adicionalmente, el uso de autoencoders permite obtener una representación compacta y de baja dimensionalidad de las señales EMG mediante un vector latente, este conserva la información más relevante tras la codificación [6, 7]. Esta representación optimizada facilita la clasificación de los movimientos, ya que permite reducir la redundancia en los datos, mejorando la precisión del modelo [8]. Al integrar estos enfoques, las redes neuronales pueden aprender a clasificar las señales EMG con mayor eficiencia y utilizarlas para controlar de manera precisa el movimiento de la silla de ruedas.

En la actualidad, se han desarrollado investigaciones con el objetivo de mejorar la precisión en el reconocimiento de patrones característicos de señales EMG. A continuación, se presentan los trabajos más relevantes de los últimos 8 años, en la adquisición de señales EMG y el entrenamiento de Redes Neuronales Recurrentes (RNN, por sus siglas en inglés) con diversas arquitecturas como el modelo de memoria a largo-corto plazo (LSTM, por sus siglas en inglés), el cual posee una capacidad de retener información de secuencias temporales [9]. El modelo de Unidad Recurrente Cerrada (GRU, por sus siglas en inglés) simplifica la estructura permitiendo reducir la carga computacional y, en contextos específicos presenta un rendimiento comparable o superior al de LSTM [10].

1.1. Antecedentes

En la bibliografía consultada, se han presentado diversas metodologías para la adquisición y clasificación de señales EMG, variando en los músculos involucrados, tipo de electrodos, filtros, características extraídas, y modelos de redes neuronales utilizadas.

En [11] se evaluaron 16 modelos basados en redes recurrentes, con 8 configuraciones LSTM y 8 para GRU, para clasificar gestos de la mano a partir de señales EMG preprocesadas mediante envolventes. Para evaluar el desempeño de los modelos, las muestras de cada movimiento de manos fueron divididas en conjuntos de entrenamiento (50%), validación (16.7%) y prueba (33.3%). La mejor precisión de validación fue de 89.5%, obtenida con una red bidireccional LSTM. Sin embargo, se observó sensibilidad en los modelos GRU a la tasa de aprendizaje, y se sugiere el uso de arquitecturas más profundas para mejorar el rendimiento.

Por su parte, [12] propuso una ANN para clasificar seis movimientos de dedos utilizando señales EMG del músculo flexor superficial de los dedos con una precisión del 95.52%. El modelo utilizado consistió en una ANN diseñada e implementada con la caja de herramientas de redes neuronales de *Matlab*®[®], que estimó la categoría de confirmación 0 o 1 para cada tipo de movimiento. Aunque se demostró eficacia con una arquitectura simple, la base de datos fue limitada a seis sujetos, lo cual restringe la generalización del modelo.

Posteriormente, en [13], se compararon modelos FFNN, LSTM y GRU usando dos bases de datos: DualMyo (1 sujeto, 8 gestos) y NinaPro DB5 (10 sujetos, 53 gestos). Se lograron precisiones de hasta 95% y 91%, respectivamente. Aunque los modelos recurrentes ofrecieron menor tiempo de entrenamiento, se señala la necesidad de optimización para su implementación en sistemas embebidos, así como el incremento de la base de datos con más sujetos.

El estudio de [14] utilizó Bi-LSTM sobre gráficos instantáneos de EMG de alta densidad, logrando una precisión del 98.57%. El uso de redes profundas permitió una mejor representación de las señales; no obstante, se enfoca en un tipo específico de representación gráfica que no es común en sistemas de bajo costo o tiempo real.

Más tarde, en [5] se empleó variantes de RNN como Peephole LSTM y GRU para señales EMG del antebrazo y pierna, alcanzando hasta un 99.6% de precisión. Se destaca la eficiencia de GRU en velocidad y simplicidad frente a modelos más complejos. Sin embargo, los experimentos fueron realizados sobre seis bases de datos específicas, lo que podría limitar la comparación directa entre modelos.

Una propuesta aplicada al control de una prótesis de mano se muestra en [15], donde se usaron cuatro electrodos de superficie y una red LSTM-RNN. El 90% de los conjuntos de datos se utilizaron para el entrenamiento y el 10% para la validación, logrando una precisión promedio de 93.79% y 90.7% respectivamente. Si bien el sistema fue probado en tiempo real, se sugiere aumentar el número de canales EMG para mejorar la precisión.

En [16], se utilizó una arquitectura LSTM de dos capas con probabilidad de abandono del 30%. Aunque se aplicaron técnicas de ajuste de tasa de aprendizaje para el optimizador NADAM, el número de sujetos en la base de datos es reducido, provocando menor representatividad.

El modelo Bi-RM propuesto por [17] para análisis de campos visuales mostró menor error de predicción con Bi-LSTM, aunque su rendimiento se vio afectado por la tasa de falsos negativos y pérdida de fijación. La aplicación de este estudio en señales EMG es limitada.

Finalmente, [18] evaluó tres variantes de RNN (GRU, LSTM, Bi-RNN) para señales EMG asociadas a cinco movimientos, logrando precisiones superiores al 96%. Además, se utilizó un filtro Butterworth y segmentación con ventanas de 250 ms. Se destaca que una cuidadosa selección de

hiperparámetros es clave para evitar caídas en la precisión durante fases experimentales posteriores.

Tabla 1.1: Resumen de modelos de ANN en bibliografía consultada.

Autor	Año	Técnica	Aplicación	Personas que conforman la base de datos	Cantidad de movimientos realizados	Precisión
[11]	2018	BILSTM-AT-LR	Clasificación de gestos con la mano a partir de señales EMG	40 sujetos sanos	50 movimientos de la mano	89.5 %
[12]	2019	Feed forward ANN	Clasificación de 6 tipos diferentes de movimientos de los dedos	6 personas sanas	7 movimientos	95.52 %
[13]	2019	LSTM, RNN, GRU	Clasificación de gestos de la mano por señales EMG.	10 sujetos / 1 sujeto	53 movimientos / 8 movimientos	91 % / 95 %
[14]	2020	LSTM, GRU, Bi-LTSM y Bi-GRU	Clasificar gestos de las manos a través del gráfico instantáneo de electromiografía de alta densidad.	18 sujetos	8 movimientos	LSTM 96.99 %, GRU 97.06 %, Bi-LTSM 98.57 %, Bi-GRU 98.08 %
[5]	2021	LSTM, GRU, CP-LSTM	Clasificar gestos de señales EMG	1 sujeto	8 movimientos	99.6 %
[15]	2023	LSTM-RNN	Clasificación de movimientos de la mano	1 sujeto	5 movimientos	90.7 %
[16]	2023	LSTM	Predicción cinética y cinemática de miembros inferiores humanos	6 sujetos	5 acciones	97.25 %
[18]	2024	LSTM-RNN, GRU-RNN, Bi-RNN	Clasificación de señales EMG asociadas a cinco movimientos de la extremidad superior derecha	9 sujetos sanos	4 músuclos	GRU 96.38 %, LTSM 98.46 %, Bi-RNN 97.63 %

El uso de señales EMG en conjunto con algoritmos de inteligencia artificial ha cobrado relevancia en el desarrollo de sistemas de reconocimiento de gestos, especialmente en el ámbito de las tecnologías asistivas y el control de prótesis. Con el objetivo de explorar las principales líneas de

investigación en esta área, se realizó un análisis bibliométrico a partir de la estrategia de búsqueda: (*emg OR .electromyographies signals. OR "myoelectric signals"*) AND (*classification OR "pattern recognition. OR "gesture recognition"*) AND (*"neural network. OR "deep learning"*) OR (*.assistive technology. OR .assistive control"*). El resultado se visualiza en la figura 1.1, una red de coocurrencia de términos, la cual permite identificar los principales conceptos.

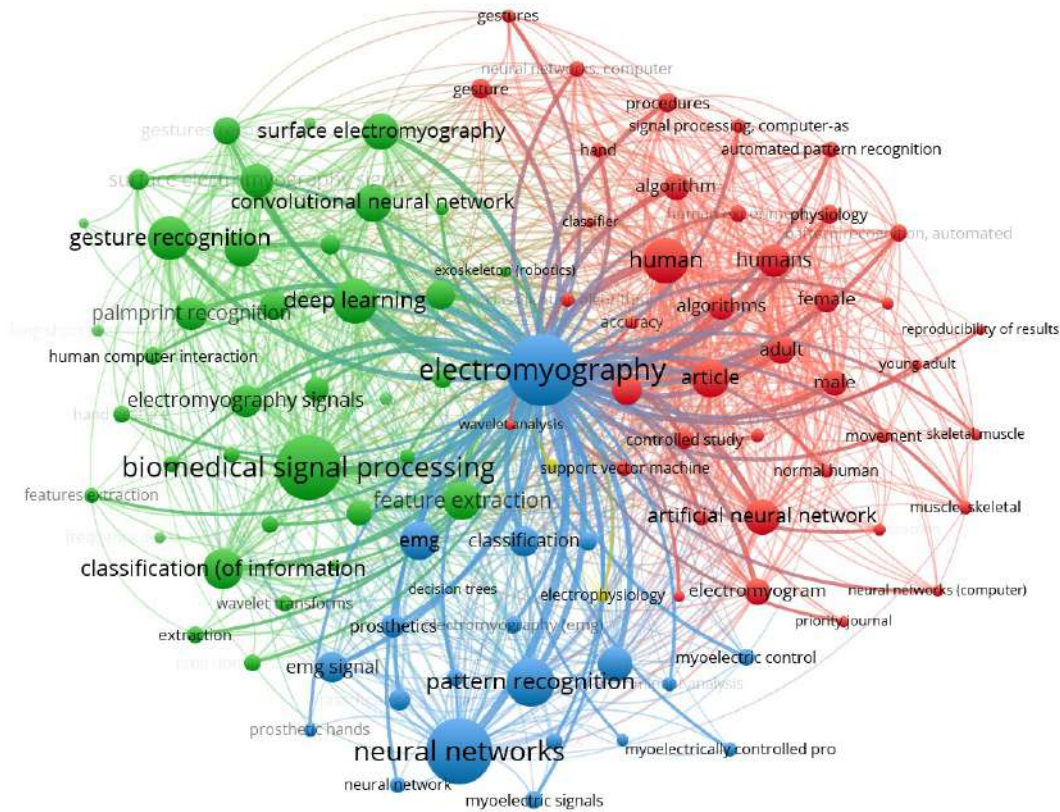


Figura 1.1: Red bibliométrica.

La red presenta tres clústeres definidos. El verde agrupa términos asociados al procesamiento de señales biomédicas y aprendizaje profundo, destacando conceptos como *surface electromyography*, *gesture recognition*, *deep learning*, *feature extraction* y *biomedical signal processing*. Refleja el enfoque actual de aplicar arquitecturas de redes neuronales profundas, para extraer características relevantes de las señales mioeléctricas.

El clúster azul se centra en el reconocimiento de patrones y el uso de redes neuronales tradicionales, incluyendo términos como *pattern recognition*, *neural networks*, *myoelectric signals* y *classification*. Este grupo resalta la evolución desde métodos clásicos hacia modelos más complejos y robustos, lo que refuerza la viabilidad de emplear técnicas como redes neuronales profundas y autoencoders para interpretar señales EMG.

Por último, el clúster rojo representa el contexto experimental y clínico, donde términos *human*, *adult*, *female*, *controlled study*. Este grupo evidencia la orientación aplicada de esta tecnología, especialmente en entornos médicos y de rehabilitación.

La red bibliométrica permite fundamentar la importancia y actualidad de los sistemas de clasificación de gestos basados en señales EMG. Sin embargo, a pesar del creciente interés por aplicar técnicas de aprendizaje profundo en este campo, no se han encontrado métodos que utilicen autoencoders para la extracción de representaciones latentes en tareas de clasificación de gestos mioeléctricos. La literatura muestra una escasa exploración del uso de redes neuronales densas como clasificadores directos de señales EMG procesadas, especialmente para tareas multiclase con cinco gestos diferentes. Esta carencia evidencia una oportunidad de innovación metodológica, donde el uso de vectores latentes obtenidos a través de codificadores automáticos puede optimizar el desempeño del sistema de reconocimiento, reduciendo la dimensionalidad de los datos.

En el presente proyecto se propone un circuito de bioinstrumentación para la adquisición de señales EMG, registradas a partir de cinco movimientos voluntarios de la mano derecha realizados por 20 personas sanas. Esta base de datos, que integra tres canales musculares y 20 repeticiones por clase, representa un conjunto de datos amplio en comparación con investigaciones similares, proporcionando una base sólida para el entrenamiento y validación de modelos de aprendizaje profundo. Las señales obtenidas serán clasificadas mediante redes neuronales recurrentes del tipo LSTM y GRU, empleando como entrada un vector latente extraído por un autoencoder. Este enfoque permite reducir la dimensionalidad y resaltar características esenciales de las señales, con el objetivo de lograr una precisión superior al 90% en la clasificación de los cinco movimientos.

1.2. Justificación

En el mundo, cerca de 28 millones de personas requieren de una silla de ruedas [19]. En México, de acuerdo al último censo realizado por el Instituto de Estadística y Geografía (INEGI), en 2015 existieron 780 mil personas amputadas, y por cada 75 mil personas que no poseen una extremidad solo el 10 % cuentan con un dispositivo protésico [20].

Para el año 2020, el INEGI dio a conocer que cerca de 3 millones de personas con discapacidad tienen dificultad para caminar, subir y bajar escaleras [1], por lo que requieren de dispositivos de asistencia como la silla de ruedas, lo que provoca que la comercialización de estas sea de alta demanda dentro del mercado [21, 22]. De igual manera, existen diversas enfermedades neurológicas tales como las lesiones en la médula espinal, y las afecciones cerebrovasculares, que afectan a un porcentaje significativo de la población disminuyendo total o parcialmente la capacidad motora [23].

A pesar de la utilidad de las sillas de ruedas, su manejo manual puede representar un desafío para muchas personas, ya que suelen ser pesadas, requieren esfuerzo físico constante y pueden generar fatiga en el tren superior. Además, el acceso a ciertas zonas con desniveles, rampas empinadas o superficies irregulares puede ser complicado, limitando la movilidad y autonomía del usuario. Ante estas estadísticas, un dispositivo de control de dirección para sillas de ruedas basado en movimientos de la mano, surge como una alternativa innovadora, permitiendo a los usuarios desplazarse de manera eficiente y reduciendo la carga física asociada al uso de sillas convencionales.

Para ello, es fundamental contar con un sistema de adquisición de señales EMG, seguido de un proceso de filtrado, procesamiento y clasificación con una eficiencia superior al 90 %, asegurando así un control preciso y fiable. La correcta clasificación de las señales es indispensable para garantizar un control eficiente, seguro y adaptable a las necesidades del usuario del usuario, permitiéndole operar el dispositivo de asistencia sin ejercer una fuerza excesiva que genere fatiga muscular en el tren superior.

Este trabajo sienta las bases para la implementación de un sistema embebido en una silla de ruedas instrumentada con motores, permitiendo la integración del modelo de clasificación desarrollado. Como resultado final en trabajos futuros, se busca obtener una silla de ruedas capaz de ejecutar cinco acciones: avanzar, retroceder, girar a la izquierda, girar a la derecha y permanecer estática, donde cada acción está asociada a movimientos de la mano: gesto de puño, flexión del pulgar, descanso, extensión de los dedos y flexión de los 3 dedos medios.

1.3. Descripción del problema

El equipo utilizado con mayor frecuencia por las personas que tienen limitaciones para caminar es una silla de ruedas; existen diversos tipos, las convencionales y las automatizadas. Las primeras al ser totalmente mecánicas, propician un desgaste físico, causando agotamiento e incomodidad al usuario, limitando a su vez el recorrer amplias distancias, y excluyendo el uso de estas a las personas que poseen poca resistencia física. Por otro lado, las sillas automatizadas proveen de mayor independencia del usuario lo que mejora su calidad de vida, sin embargo, éstas suelen ser más voluminosas y costosas lo que las convierte en un producto que no está al alcance de toda la población [24].

En la instrumentación biomédica, la electromiografía ha sido ampliamente utilizada para el control de prótesis mioeléctricas por la información muscular que contienen [25, 26]. Dado el éxito de estas aplicaciones en prótesis, surge la oportunidad de extender el uso de las señales EMG hacia otros dispositivos de asistencia, como las sillas de ruedas. Por lo que, ante esta problemática se propone la implementación de un sistema de clasificación de señales musculares para el control de dirección de una silla de ruedas que permita brindar independencia y desplazamiento sin desgaste físico al usuario, haciendo uso de técnicas de inteligencia artificial para una clasificación eficiente de las señales musculares.

Para ello, es indispensable desarrollar un sistema robusto de adquisición, filtrado, procesamiento y clasificación de señales EMG, que garantice una alta precisión en la identificación de los movimientos musculares y su traducción en comandos de control. Diversos trabajos han desarrollado algoritmos capaces de interpretar estas señales y convertirlas en comandos precisos para la manipulación de prótesis [15, 16]. No obstante, hasta donde se tiene conocimiento, la aplicación de estas técnicas para el control de sillas de ruedas sigue siendo un área poco explorada.

1.4. Hipótesis

La identificación de patrones específicos a través del procesamiento de señales EMG y redes recurrentes durante la actividad muscular permite la detección de 5 movimientos correspondientes a 5 acciones hechas por el usuario con una eficiencia de clasificación mínima del 90 %.

1.5. Objetivo General

Desarrollar una interfaz gráfica para visualizar la clasificación de señales EMG de la mano mediante redes neuronales, facilitando su interpretación y análisis para el control de dispositivos de asistencia.

1.6. Objetivos Específicos

- Diseñar un circuito de bioinstrumentación para la obtención de señales EMG.
- Crear una base de datos de señales de EMG obtenidas de 20 personas realizando cinco movimientos característicos de la mano.
- Procesar las señales EMG para la reducción de ruido y disminución de carga computacional integrando técnicas de segmentación y ventaneo.
- Desarrollar un autoencoder para extraer el vector latente con las características esenciales de las señales EMG, permitiendo su reconstrucción.
- Implementar un modelo de redes neuronales recurrentes (LSTM y GRU) para clasificar las señales EMG reconstruidas a partir del vector latente.
- Evaluar el rendimiento del clasificador con métricas como exactitud, sensibilidad, especificidad y eficiencia.
- Integrar el algoritmo de clasificación de señales EMG en una interfaz gráfica.

Fundamentación teórica

En la presente sección se detallan las técnicas que serán utilizadas en el desarrollo del proyecto.

2.1. Contracción Muscular

Existen tres tipos de músculos en el cuerpo humano, los músculos esqueléticos, cardíacos y lisos. El sistema nervioso central es responsable de los músculos esqueléticos, estos músculos generan fuerza y realizan movimientos de traslación del cuerpo, constituyendo gran parte de la masa muscular [27]. Su unidad estructural es la fibra muscular [28], estas fibras se contraen para producir el movimiento o fuerza, y están formadas por la miofibra muscular o célula muscular. A través de la red neuronal, la señal que se genera por la corteza motora llega al músculo esquelético, la red neuronal está formada por neuronas motoras (MU) interconectadas [2].

Después de que el impulso nervioso viaja por la médula espinal y activa la fibra muscular, se lleva a cabo el proceso electroquímico, en donde por medio de un estímulo se abren los canales sodio-potasio, provocando una respuesta activa de las membranas excitables en las fibras musculares y en las fibras nerviosas [2]. Las acciones polarizantes y despolarizantes de la apertura de los canales de sodio-potasio producen la señal mioeléctrica [29].

2.2. Adquisición de señal EMG

La electromiografía es una técnica para el registro de actividad eléctrica de los músculos esqueléticos [3], esta permite la detección y evaluación de perturbaciones fisiológicas en las membranas de las fibras de los músculos, probando todo el sistema motor [30, 31].

La obtención de señales EMG es sencilla, sin embargo, es necesaria la implementación de sistemas robustos para su interpretación cuantitativa [32]. Para la obtención de estas bioseñales, se utiliza una técnica no invasiva llamada electromiografía de superficie en la que se utilizan pequeños discos metálicos llamados electrodos bipolares de superficie de plata/cloruro de plata (Ag/AgCl) colocados en la piel del paciente [33].

El principio en el que se sustenta esta técnica es que el músculo al contraerse, emite pequeñas descargas eléctricas que son detectadas por el electromiógrafo. Por el análisis de esta actividad eléctrica producida, se puede identificar alguna anomalía existente por lesión o alteración patológica [31].

En la figura 2.1 se observa la conexión de la neurona con el músculo, mediante el axón, el impulso nervioso es conducido desde el soma hasta las fibras musculares y transmite la información desde el sistema nervioso central hasta el músculo esquelético para realizar un movimiento [34].

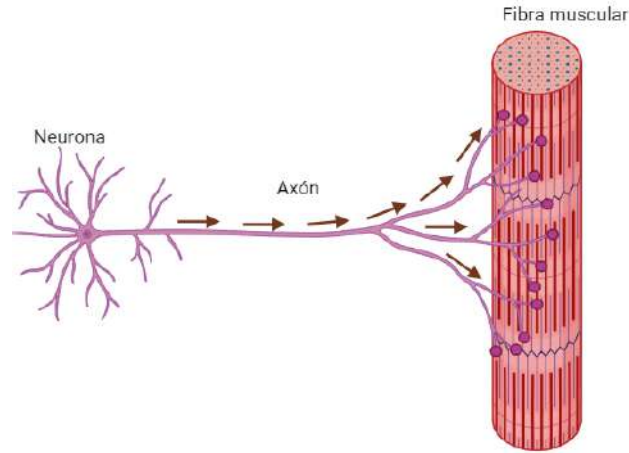


Figura 2.1: Conexión de la neurona a la fibra muscular por medio del axón.

El control mioeléctrico se basa en la conexión entre la neurona y el músculo. Para este control, es recomendable utilizar más de una vía que permita la transmisión de señales mioeléctricas de un electrodo a un sistema de adquisición (canal), ya que el movimiento generado es con un conjunto de músculos. El uso de canales dependerá del tipo de acciones que se realizarán y la aplicación que tenga la investigación. Trabajos relacionados han utilizado cuatro, seis u ocho canales para su adquisición [2].

Existen tres derivaciones de electrodos comúnmente utilizadas: la configuración bipolar, la monopolar y la laplaciana. En la configuración monopolar se mide la diferencia existente entre el electrodo de medición (colocado en el músculo) y en electrodo de referencia (tierra) [35]. En la derivación bipolar de un canal señal EMG es la diferencia de voltaje entre un par de electrodos de superficie de registro alineados a lo largo de la superficie de la piel del músculo [36]. Por otro lado, la configuración laplaciana utiliza electrodos circundantes y un electrodo de superficie central [37, 38]. La derivación de EMG bipolares son los preferidos en las interfaces EMG, ya que son más tolerantes al ruido que los monopolares [39]. Es necesario garantizar una calidad de la señal alta, para obtener una alta precisión de clasificación, por lo que seleccionar el número de canales y la disposición, es una fundamental [40].

2.3. Señal EMG

Las señales EMG registran la cantidad de potencial eléctrico que se obtiene de las células musculares cuando estas son contraídas o están en reposo [41].

Una señal en crudo de EMG, es la señal obtenida sin filtrar y sin procesar que detecta el potencial de acción de unidad motora [30]. Este tipo de señales tienen amplitudes pico a pico de 0-10 mV con una frecuencia de banda que va de 0 Hz a 1 kHz [2], siendo el rango de frecuencia de la señal EMG de 10 a 500 Hz [42]. Sin embargo, autores han sugerido que el rango útil es de 20 a 500

Hz [43, 44]. Por lo que en cualquier frecuencia por encima de 500 Hz y debajo de 10 Hz se encuentra ruido producido por las capas de la piel o por el movimiento de los cables de los electrodos [40, 44].

Para la adquisición de la señal EMG, es necesaria su amplificación, filtrado y conversión a señal digital [45], a continuación se explican estas etapas.

2.4. Amplificación

Por lo general, la amplitud de una señal EMG se encuentra por debajo de los 10 mV [2], esto provoca que sea propensa a la perturbación. Es por esto que, se amplifica normalmente de 100 a 5000 veces con un amplificador de instrumentación, preferiblemente lo más cerca posible de los electrodos de grabación, y posteriormente se filtra [46].

2.5. Filtrado

El filtrado consiste en obtener el rango de información útil de la señal, en esta etapa se elimina el ruido y artefactos que la señal pueda contener [47]. Los filtros se pueden clasificar con base a su respuesta en frecuencia, es decir, de acuerdo al paso de señales con determinadas bandas de frecuencia y al bloqueo de señales con otras bandas de frecuencia [48].

Dentro de estos tipos de filtro se encuentran el filtro pasa altas y el filtro pasa-bajas. El primero permite el paso de las frecuencias más altas que una frecuencia de corte específica, atenuando las más bajas. Mientras que un filtro pasa bajas permite el paso de las frecuencias más bajas que una frecuencia de corte específicas atenuando las más altas [47].

Cuando se combina un filtro pasa-bajas y un filtro pasa-altas, a la región fuera de la banda de paso se conoce como rechazabanda, obteniendo de esta manera el filtro rechazabanda o filtro Notch. Este permite el paso de frecuencias altas y frecuencias bajas, sin embargo, atenúa cualquier señal que tenga una frecuencia entre las dos frecuencias de corte [49].

El filtro pasa-altas es crucial por la presencia de espectros superpuestos como los movimientos del cuerpo, estos pueden ser de hasta 20 Hz, y los artefactos causados por cables de 50Hz. Para filtrar estos artefactos la frecuencia de corte dependerá de la aplicación así como en el músculo que se realizará el estudio, sin embargo, [50] afirma que no es recomendable el uso de frecuencias de corte menores a 20 Hz por los artefactos de movimiento.

Un frecuencia de corte de filtro pasa altas que oscila entre 5 y 20 Hz es utilizado con frecuencia en estudios de EMG para eliminar las variaciones lentas de la señal causadas por el movimiento de los cables de los electrodos [39].

Sin embargo, literatura ha sugerido una frecuencia de corte pasa altas a 60 Hz para las interfaces EMG de superficie, debido a las interferencias de potencia por corriente, así como por la información no relevante sobre las velocidades de disparo de las unidades motoras activas [46, 51].

Para filtros pasa-bajas es acertado el uso de frecuencia de corte en un rango de 400 a 500 Hz, ya que el espectro EMG comprende componentes de hasta 500 Hz. Sin embargo, la energía por encima de 350 Hz es muy baja por lo que esta frecuencia de corte resulta no ser crítica [50].

2.6. Segmentación

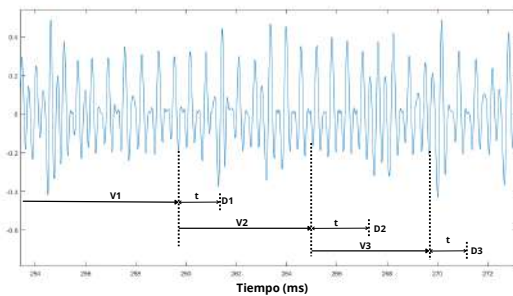
En las interfaces EMG, los datos deben estudiarse en tiempo real, por lo tanto, el análisis se realiza en segmentos de tiempo, es decir, ventanas [52, 40]. Este preprocesamiento afecta la

precisión de la clasificación y el tiempo de respuesta del sistema. Existen dos técnicas de ventaneo, las ventanas adyacentes y ventanas superpuestas [43].

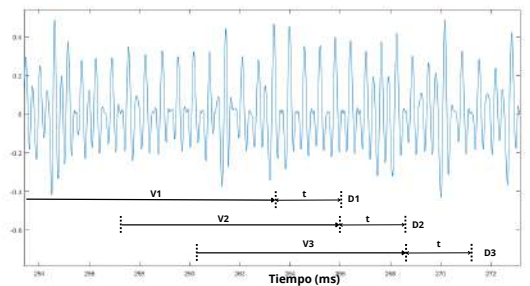
En las adyacentes, se utilizan segmentos de longitud consecutivos para el análisis y la extracción de características, en donde el tiempo suele a ser menor que la duración del segmento de tiempo debido a los procesadores de alta velocidad [53, 54].

Por otro lado, las ventanas superpuestas utilizan el tiempo de inactividad para adquirir más datos que se van a procesar, ya que cada segmento se superpone al anterior. El enfoque de ventana superpuesta es más apropiado en los sistemas de control de EMG porque produce una mejor precisión de clasificación y un retardo de controlador más constante y reduce la longitud del retardo máximo [54].

En la figura 2.2 se muestran los dos tipos de ventanas, siendo 2.2a las ventanas adyacentes y 2.2b las superpuestas.



(a) Técnica de ventanas adyacentes.



(b) Técnica de ventanas superpuestas.

Figura 2.2: Técnicas de ventaneo.

Las grandes ventanas de datos aumentan la precisión de la clasificación, no obstante, presentan la desventaja de requerir más tiempo para recopilar y procesar conjuntos de datos más grandes, provocando que la carga computacional sea mayor. Por lo tanto, es importante tener un equilibrio entre la precisión de la clasificación y las restricciones en tiempo real [39].

Un tiempo de 200 ms es el existente entre una contracción del músculo y otra, por lo que un intervalo mayor o igual a este tiempo contiene la información necesaria para estimar los estados de movimiento de una extremidad [55]. Sin embargo, con la implementación de mecanismos de preprocesamiento y una selección de características apropiado es posible una alta precisión de clasificación con segmentos inferiores a 200 ms [54].

La ecuación utilizada para obtener el número de ventanas se describe en la ecuación 2.1. Donde N es el número total de ventanas, L es la longitud total de la señal, w es el tamaño de la ventana y t es el traslape entre las ventanas, todo en número de muestras.

$$N = \left\lfloor \frac{L - w}{w - t} \right\rfloor + 1 \quad (2.1)$$

2.7. Inteligencia Artificial

La inteligencia artificial (IA) se define como la aptitud de los sistemas informáticos o programas para llevar a cabo acciones que habitualmente son atribuidas a la inteligencia humana. Estas actividades abarcan desde el aprendizaje y la adaptación hasta la resolución de problemas y la toma de decisiones. La implementación de la IA se logra mediante la utilización de algoritmos y modelos que habilitan a las máquinas para ejecutar tareas específicas de manera autónoma, sin requerir una intervención directa por parte de los seres humanos [56].

2.7.1. Aprendizaje Automático

El aprendizaje automático, surge del campo de la IA, esta rama tiene por objetivo desarrollar sistemas que aprenden automáticamente al reconocer patrones para predecir comportamientos a partir de un conjunto de datos [57].

En las técnicas convencionales del aprendizaje automático se identificó una limitada capacidad de procesamiento de datos naturales en forma bruta, ya que para el diseño de estos sistemas se requería de una ingeniería cuidadosa. Es por ello que los sistemas de aprendizaje automático hacen uso de técnicas de aprendizaje profundo, para realizar acciones como identificar objetos en imágenes o transcribir voz en texto [4].

2.7.2. Aprendizaje Profundo

El aprendizaje profundo, al igual que el aprendizaje automático, aprende a identificar patrones complejos en conjuntos de datos mediante el algoritmo de propagación inversa, sin embargo, el primero utiliza modelos más complejos con múltiples capas ocultas, como redes neuronales profundas, redes convolucionales y redes recurrentes. Este algoritmo le indica a la máquina cómo ajustar sus parámetros internos para que pueda aprender a relacionar las características de los datos de entrada con las características de la representación en cada capa [4].

Redes Neuronales Artificiales

Dentro de las herramientas más importantes del aprendizaje automático, se encuentran las Redes Neuronales Artificiales (ANN). Estas son un modelo matemático que simula la estructura y las funciones de las redes neuronales biológicas [58], esta tecnología ha sido aplicada en la resolución de problemas complejos.

Las ANNs, pretenden de emular el comportamiento y función del cerebro humano, específicamente el del sistema nervioso [59], el cual está compuesto por redes neuronales biológicas, caracterizándose de igual forma por el aprendizaje a través de la experiencia y la extracción de conocimiento genérico a partir de un conjunto de datos.

Los modelos de ANN son capaces de encontrar patrones de forma inductiva por medio de los algoritmos de aprendizaje basado en los datos existentes, cuando un conjunto de entradas se aplica a una neurona de la ANN, donde cada entrada representa la salida de otra neurona, la entrada es multiplicada por su peso correspondiente análogo al grado de conexión de la sinapsis [60]. Estas entradas ponderadas se suman para determinar el nivel de activación de la neurona.

A continuación se muestra una representación vectorial del funcionamiento básico de una neurona artificial.

$$N = x \cdot w \tag{2.2}$$

Siendo N la salida, x el vector de entrada y w el vector de pesos. La señal N se procesa por una función de activación f , para producir la señal de salida de la neurona. La figura 2.3 muestra un esquema de la Neurona artificial.

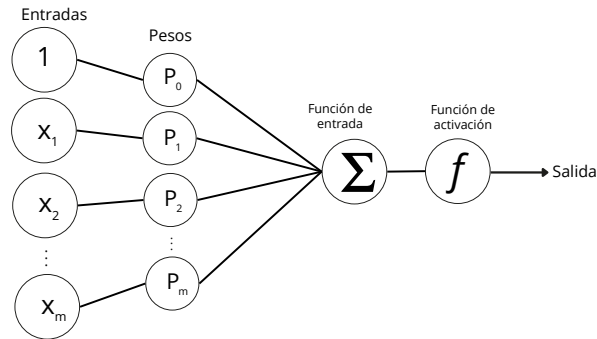


Figura 2.3: Esquema de una neurona artificial.

Parámetros e hiperparámetros de una red neuronal

Todas las redes neuronales artificiales están compuestas por procesador elemental llamada neurona, cada neurona tiene una o múltiples entradas y opcionalmente una entrada constante (sesgo), y producen salidas [61]. Las entradas se multiplican por los pesos, estos son valores que determinan la fuerza de las conexiones entre las neuronas y las salidas se calculan aplicando una función de activación a la suma ponderada de las entradas[62].

El entrenamiento consiste en ajustar los pesos y sesgos de las neuronas para que la red realice predicciones precisas en nuevos datos de entrada, es por esto que desempeñan un papel fundamental en el funcionamiento de las ANN, ya que determinan la forma en que las neuronas se conectan entre sí y cómo la red interpreta los datos [63]. Los valores de los pesos pueden ser fijos desde el inicio, o pueden encontrarse utilizando diferentes algoritmos de entrenamiento de la red [61].

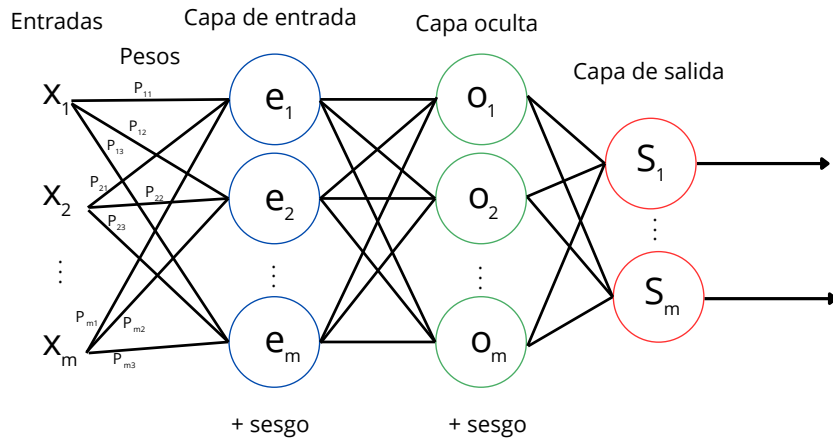


Figura 2.4: Esquema de una red neuronal y sus parámetros.

Hiperparámetros

Para la programación de redes neuronales se establecen parámetros externos de forma manual o automática durante la fase de entrenamiento. La precisión de las salidas que se obtienen de la red neuronal está afectada por los hiperparámetros [5].

Función de activación

La activación de cada neurona depende de las entradas, los pesos, sesgo y de la función de activación neuronal, esta última determina un rango que definirá el estado de activación de la neurona. Las funciones más comunes son tangente hiperbólica (\tanh), sigmoideo, y rectificación lineal (ReLU, por sus siglas en inglés) [64], en el cuadro 2.1 se muestran la función y el dominio de cada una.

Tabla 2.1: Funciones de activación obtenidas

Función	Expresión	Dominio
Tangente hiperbólica	$F(X_i) = \frac{\sinh(X_i)}{\cosh(X_i)}$	$(-1, 1)$
Sigmoide	$F(X_i) = \frac{1}{1 + e^{X_i}}$	$(0, 1)$
Lineal	$F(X_i) = \max(0, X_i)$	$[0, \infty)$

La función de activación expresada matemáticamente, define la salida de la neurona en términos de campo local inducido [65]. La salida de una neurona se calcula multiplicando las entradas de la neurona por sus pesos, para posteriormente sumar los resultados. La función de activación se aplica a la suma ponderada de las entradas para obtener la salida final de la neurona, determinando así, la forma en que las neuronas procesan la información y cómo es que la red aprende [61].

En el diseño de redes neuronales profundas, las funciones de activación juegan un papel crucial en la propagación de la información y la optimización del modelo [66]. Entre las más utilizadas se encuentran GELU, ELU y ReLU, cada una con características particulares:

Unidad lineal del error gaussiano GELU, por sus siglas en inglés, es una función de activación que ha demostrado un rendimiento superior en diversas aplicaciones. Suaviza la activación mediante una función gaussiana, lo que le otorga propiedades matemáticas como diferenciabilidad, acotación y suavidad, mejorando la estabilidad del entrenamiento [67].

Unidad lineal exponencial Por sus siglas en inglés, ELU, introduce una transformación exponencial en los valores negativos, lo que permite reducir la desaparición del gradiente y mejorar la precisión de clasificación en comparación con ReLU [68].

Épocas

Una época es el proceso en el que el conjunto de datos completos ha transitado una vez por el modelo de la red neuronal hacia adelante y hacia atrás, este proceso actualiza los pesos de la gradiente descendente [69].

Lotes

También conocido como batch size, este hiperparámetro es la distribución de los datos de entrada en un conjunto de muestras que se procesan de manera paralela e independiente [69].

Retropropagación

La retropropagación es un algoritmo fundamental en el entrenamiento de RNN, ya que permite calcular de forma eficiente los gradientes de una función compuesta con respecto a sus variables de entrada, lo que facilita la optimización de los pesos de la red neuronal durante el proceso de aprendizaje. Este algoritmo, propaga el error entre el valor actual y el resultado de la red neuronal. Al proporcionar un patrón de entrada, la salida es comparada con un patrón objetivo para calcular el error de cada neurona en la salida. Este error se propaga hacia atrás estableciendo un sistema de control de bucle cerrado [70].

Dentro del contexto de las redes neuronales, la optimización se centra en el rendimiento de la red, con el fin de encontrar un conjunto de pesos que minimice el error. No obstante, esta con frecuencia es intratable y no depende del conjunto de entrenamiento, sino, del de prueba. Es por esto que para optimizar el rendimiento se realiza de manera indirecta, mediante otra función llamada función de costo. Esta función de costo cuantifica la pérdida en la que se incurre para cierto valor de los parámetros. Por lo general, el costo se minimiza sobre el conjunto de entrenamiento [71].

Existen diferentes técnicas efectivas de optimización como el descenso por gradiente y Adam.

▪ **Descenso de gradiente**

El descenso de gradiente es un algoritmo de optimización utilizado en el entrenamiento de ANN, en el que se desea encontrar un mínimo de la función de costo [70].

La optimización de gradiente de una red neuronal se produce mediante el entrenamiento de descenso de gradiente estocástico. En este proceso, la máquina modifica los pesos internos de las conexiones entre capas para reducir el error, para ello, calcula un vector de gradiente, y se ajusta en su dirección opuesta buscando un mínimo donde el error promedio sea bajo [72].

Mediante el entrenamiento de descenso de gradiente estocástico, se modifican los pesos de las conexiones entre las capas para reducir el error, se calcula un vector de gradiente y se ajusta

en dirección opuesta para buscar donde el promedio sea mínimo, este proceso se conoce como optimización de gradiente [4].

▪ Método ADAM

El método de Estimación de Momento Adaptativo (ADAM, por sus siglas en inglés) es un algoritmo de optimización estocástica para el entrenamiento de redes neuronales, requiere de dos tipos de gradientes para calcular la tasa de aprendizaje: los gradientes de primer momento y los gradientes de segundo momento. Estos dos son utilizados para calcular una tasa de aprendizaje adaptativa para cada parámetro de la red neuronal [73]. La tasa de aprendizaje controla la velocidad a la que se ajustan los pesos de una red neuronal durante el entrenamiento.

Redes Neuronales Recurrentes

Las RNNs incorporan un estado recurrente oculto cuyo valor depende del término anterior en el tiempo [11].

En los cuadros 2.2 y 2.3 se definen las variables y subíndices empleados en la obtención de ecuaciones para modelos de redes neuronales.

Tabla 2.2: Definición de variables empleadas en modelos de redes neuronales.

Variable	Definición
t	Tiempo
ht	Salida en paso del tiempo
ϕ	Función de activación
W	Matriz de peso asociada con variables de entrada
U	Matriz de peso asociada con variables de salida
V	Matriz de peso asociada con variables de memoria
σ	Activación sigmoide
\tanh	Activación tangencial hiperbólica
ft	Puerta de olvido que controla la cantidad de memoria existente transferida
c'_t	Nuevo contenido de memoria modulado
z_t	Compuerta de actualización
r_t	Puerta de restablecimiento en el tiempo t
h'_t	Salida candidata
r'_t	Puerta de restablecimiento

Tabla 2.3: Definición de subíndices empleadas en modelos de redes neuronales

Subíndice	Definición
f	Puerta de olvido
o	Puerta de salida
i	Puerta de entrada

Para una secuencia de entrada $x = [x_1, x_2, \dots, x_t]$ alimentada a una unidad recurrente, la salida en cada paso de tiempo t , h_t se calcula como:

$$h_t = \begin{cases} 0, & t = 0 \\ \phi(h_{t-1}, x_t), & \text{otro caso} \end{cases} \quad (2.3)$$

Donde ϕ es una función de activación. El mapeo, es de la forma

$$h_t = \phi(Wx_t + Uh_{t-1})$$

La salida de una unidad recurrente en el tiempo t es una distribución de probabilidad sobre el siguiente elemento de la secuencia. El entrenamiento de la RNN mediante la propagación hacia atrás resulta complicado por problemas con el cálculo del gradiente, principalmente, gradientes que desaparecen [74].

Para presentar una solución ante el gradiente de fuga producido durante la retropropagación en el tiempo, se introdujeron dos variaciones principales de unidades recurrentes, LSTM, GRU [75, 76].

Arquitectura LSTM

Una unidad LSTM utiliza tres puertas multiplicativas separadas, de entrada (i), olvido (f) y salida (o), cada una tiene una función específica. Mantiene una celda de memoria interna para la cual las acciones de escritura, restablecimiento y lectura se rigen por las tres puertas [11].

Mediante la puerta de entrada i se especifica la cantidad de información nueva permitida en la celda [5].

La puerta del olvido, f es responsable de decidir cuándo olvidar el material relevante para el estado interno, mientras que la puerta de salida es responsable de controlar qué detalles pueden ir a la salida [77].

La puerta de salida o es la responsable del control sobre qué datos pueden ir a la salida.

A continuación, las activaciones tangentes sigmoide e hiperbólica se denotan como σ y \tanh , respectivamente, W , U y V denotan matrices de peso asociadas con variables de entrada, salida y memoria, respectivamente.

La salida LSTM como tiempo t se calcula como:

$$h_t = o_t \cdot \tanh(c_t) \quad (2.4)$$

Donde o_t es la variable de la puerta de salida en el tiempo t modulando la cantidad de exposición a la memoria c_t y se obtiene

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + V_o c_t) \quad (2.5)$$

La variable de puerta de memoria en el tiempo t , denotada como c_t se calcula:

$$c_t = f_t \cdot c_{t-1} + i_t \cdot c'_t \quad (2.6)$$

Donde f_t es la variable de puerta de olvido que controla la cantidad de memoria existente transferida y c'_t es el nuevo contenido de memoria modulado por la variable de puerta de entrada i_t .

$$c'_t = \tanh(W_c x_t + U_c h_{t-1}) \quad (2.7)$$

En la figura 2.5 se muestra una imagen de como está estructurada la arquitectura LSTM.

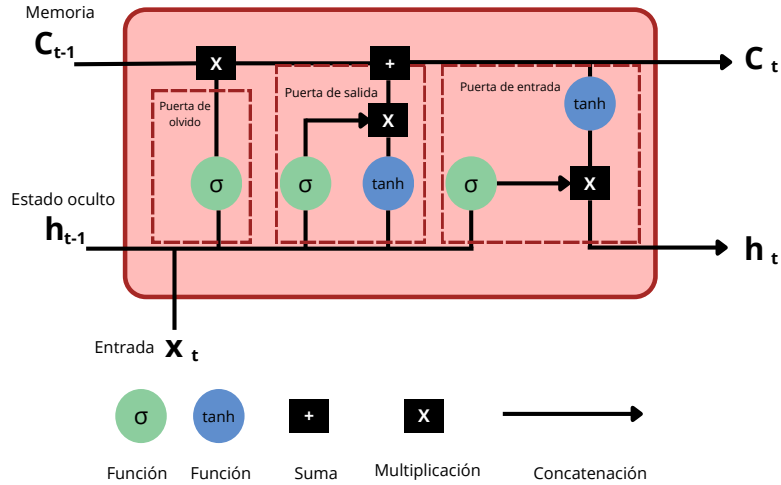


Figura 2.5: Arquitectura LSTM.

Arquitectura GRU

En capacidad de aprendizaje, la LSTM requiere un gran tiempo de cálculo. Por este motivo se creó GRU para permitir que cada unidad recurrente almacene dependencias de forma adaptativa en varias escalas de tiempo [5].

Al igual que la LSTM, la GRU utiliza unidades de compuerta para regular el flujo de información dentro de la unidad, pero sin celdas de memoria separadas.

La salida de una unidad GRU en el momento t es:

$$h_t = (1 - z_t) \cdot (h_{t-1} + z_t \cdot h'_t) \quad (2.8)$$

Donde z_t es la variable de la compuerta de actualización que controla el alcance de la actualización de la salida GRU en cada paso de tiempo t .

$$z_t = \sigma(W_z x_t + U_z h_{t-1}) \quad (2.9)$$

$$h'_t = \tanh(W_o x_t + U_o (r_t \cdot h_{t-1})) \quad (2.10)$$

Donde r_t es la variable de puerta de restablecimiento en el tiempo t , si es cercano a cero borra la contribución del estado calculado previamente en h'_t . La variable de puerta de restablecimiento se calcula:

$$r_t = \sigma(W_r x_t + U_r h_{t-1}) \quad (2.11)$$

En la figura 2.6 se muestra una imagen de como está estructurada la arquitectura GRU.

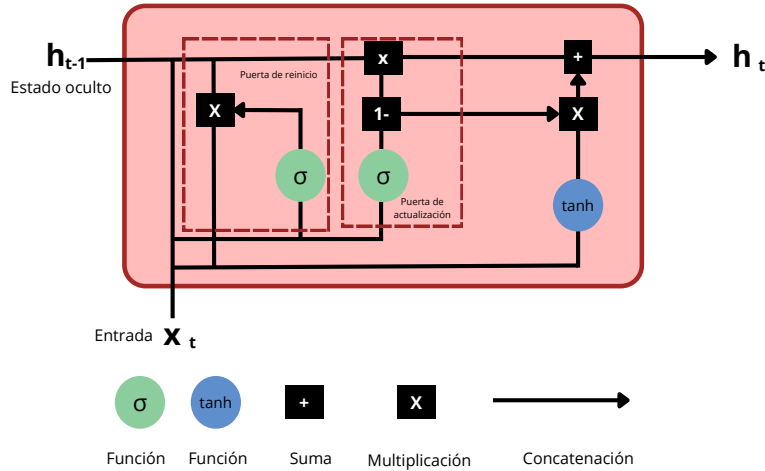


Figura 2.6: Arquitectura GRU.

Capas densas

Las redes neuronales profundas tienen una o más capas densas al final de la arquitectura, después de las capas convolucionales o recurrentes. Estas capas densas aprenden las relaciones complejas entre las características extraídas por las capas anteriores y las etiquetas de salida [78].

A diferencia de las capas convolucionales o recurrentes, las capas densas no tienen una estructura espacial o temporal, sino que cada neurona está conectada a todas las neuronas de la capa anterior [79]. Lo cual permite que la red aprenda relaciones más generales entre las características de entrada y salida.

2.7.3. Autoencoder

Un autoencoder es un tipo de red neuronal artificial que se utiliza para el aprendizaje no supervisado, diseñada para codificar y decodificar datos de entrada. Su principal objetivo es aprender representaciones informativas de los datos de una manera eficiente [80].

Los autoencoders constan de tres componentes principales: un codificador (encoder) el cual transforma los datos de entrada en una representación de características de menor dimensionalidad, un espacio de características latentes, y un decodificador (decoder) que reconstruye la entrada original a partir de esta representación [81]. Las capas LSTM y GRU aprenden las representaciones latentes de la señal, utilizando activaciones como ELU, GELU y RELU, para mejorar el aprendizaje de las características temporales [82].

Se han utilizado diversas arquitecturas de RNN, incluyendo GRU, LSTM, LSTM apilados y GRU apilados, en la extracción de características en aplicaciones como la detección de anomalías en datos de audio de compresores y la detección de intrusiones en redes informáticas [83, 84].

Funciones de pérdida

Las funciones de pérdida desempeñan un papel fundamental en el entrenamiento de los autoencoders ya que guían su proceso de optimización y aprendizaje [85, 86].

La función de pérdida mide la diferencia entre la entrada y la salida reconstruida, con el objetivo de minimizar esta diferencia.

Algunos tipos de funciones de pérdida se describen a continuación.

Error Cuadrático Medio: Es el promedio de los cuadrados de las diferencias entre los valores predichos y los valores reales [87, 88].

Matemáticamente, se expresa como:

$$ECM = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (2.12)$$

Raíz Cuadrada del Error Cuadrático Medio: Es la raíz cuadrada del error cuadrático medio [87].

Se define como:

$$RECM = \sqrt{ECM} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (2.13)$$

Error Medio Absoluto : Es el promedio de los valores absolutos de las diferencias entre los valores predichos y los reales [87, 88].

Matemáticamente:

$$EMA = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (2.14)$$

Error Medio Absoluto Porcentual : Es el promedio de los valores absolutos de los errores porcentuales [87, 88].

Se calcula como:

$$EMAP = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \times 100 \quad (2.15)$$

Estas funciones de error se utilizan ampliamente en aprendizaje de máquina y minería de datos para evaluar el desempeño de los modelos predictivos. La elección de la función de error depende del problema específico y de las propiedades deseadas del modelo.

2.8. Validación del modelo

Existen diversas técnicas para validar los métodos de regresión, como el uso de técnicas de validación cruzada. El método *hold-out* separa los datos en dos subconjuntos. El primer subconjunto se utiliza para entrenar el modelo y el segundo, creado con datos que no han sido utilizados en el entrenamiento, se emplea para realizar el test de validación [89, 90]. El modelo se crea solo con el subconjunto de entrenamiento, posteriormente, se generan datos de salida que son comparados con el subconjunto de validación.

Las estadísticas que se obtienen con los datos del subconjunto de validación son los que proporcionan la validez del método empleado en términos de error [91].

La eficiencia es calculada por un promedio de diferentes factores de k pruebas. Los falsos negativos y los falsos positivos son obtenidos en cada iteración que realiza el sistema y se utilizan para calcular: la exactitud (Ec. 2.16), la sensibilidad (Ec.2.17), la especificidad (Ec. 2.18) y la eficiencia (Ec. 2.19).

$$Exactitud = \frac{VP + VN}{VP + VN + FP + FN} \quad (2.16)$$

$$Sensibilidad = \frac{VP}{VP + FN} \quad (2.17)$$

$$Especificidad = \frac{VN}{VN + FP} \quad (2.18)$$

$$Eficiencia = \frac{Exactitud + Sensibilidad + Especificidad}{3} \quad (2.19)$$

Las variables para las ecuaciones, se definen en el cuadro 2.4.

Tabla 2.4: Definición de variables empleadas en la validación del modelo

Variable	Definición
VN	Casos verdaderos negativos
FN	Casos falsos negativos
VP	Casos verdaderos positivos
FP	Casos falsos positivos

La exactitud mide el porcentaje total de predicciones correctas (positivas y negativas) con respecto al total de casos. La sensibilidad, también llamada tasa de verdaderos positivos mide la capacidad del modelo para detectar correctamente las instancias positivas. La especificidad mide la capacidad del modelo para identificar correctamente las instancias negativas. La eficiencia es el promedio de las tres métricas anteriores.

Metodología

En esta sección se detallan las fases para el desarrollo del sistema de adquisición y el de clasificación. Estas incluyen la creación del sistema de adquisición, la creación de la base de datos, el preprocesamiento, la selección de hiperparámetros y la clasificación, que se ejecutarán usando *MATLAB R2024a* y *Python 3.10.12* en un equipo Lenovo con 8GB de RAM, un procesador Ryzen7 y una tarjeta gráfica AMD Radeon. La Figura 3.1 muestra el diagrama general del proyecto.

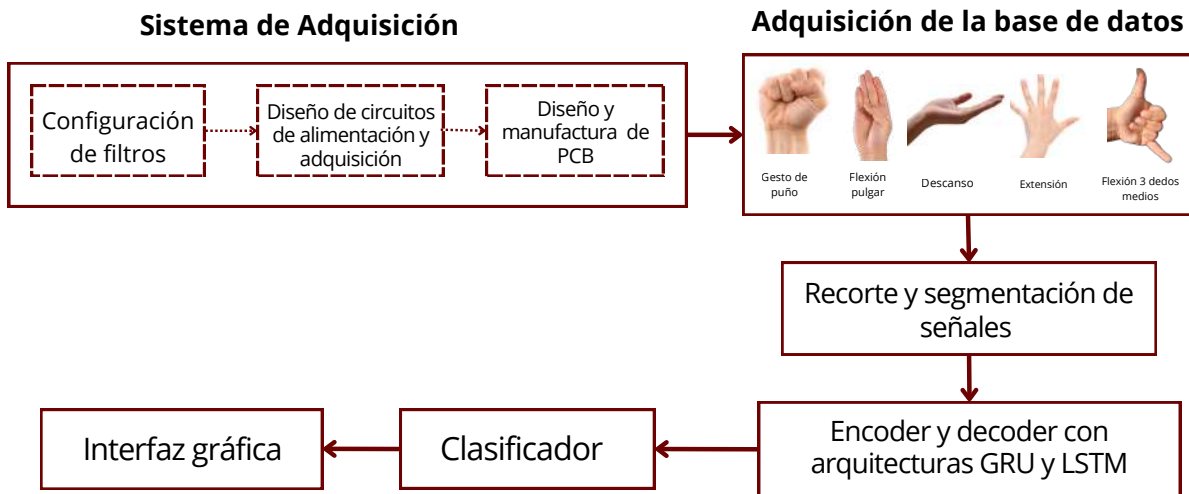


Figura 3.1: Esquema de la metodología general para el clasificador de señales EMG.

3.1. Sistema de adquisición

En esta sección se describen las etapas realizadas para la elaboración del sistema de adquisición.

3.1.1. Diseño del circuito de alimentación

La alimentación de las tres placas de adquisición se realiza mediante dos baterías recargables de ion-litio de 3.4 V conectadas en serie. Para generar las tensiones necesarias, se emplea un regulador de voltaje conmutado TC7660SCPA junto con dos reguladores 7805. En este circuito, un regulador 7805 proporciona la salida positiva de +5 V, mientras que el TC7660SCPA, en combinación con el segundo 7805, genera la tensión negativa de -5 V, asegurando una fuente de alimentación dual estable.

3.1.2. Diseño del circuito de adquisición de señales EMG

El circuito inicia con dos LMC6064 que funcionan como seguidores de voltaje para aislar la señal EMG y evitar que la impedancia de los electrodos afecte el circuito. Estos buffers permiten que la señal pase sin modificaciones de amplitud, pero con menor impedancia de salida, facilitando su procesamiento en las siguientes etapas. En la Figura 3.2 se observa el amplificador instrumental INA118U que diferencia la señal, recibida de las señales procesadas por los seguidores, aplicando una primera amplificación ajustada por la resistencia de ganancia de 24Ω ayudando a resaltar la actividad muscular sobre el ruido de fondo.

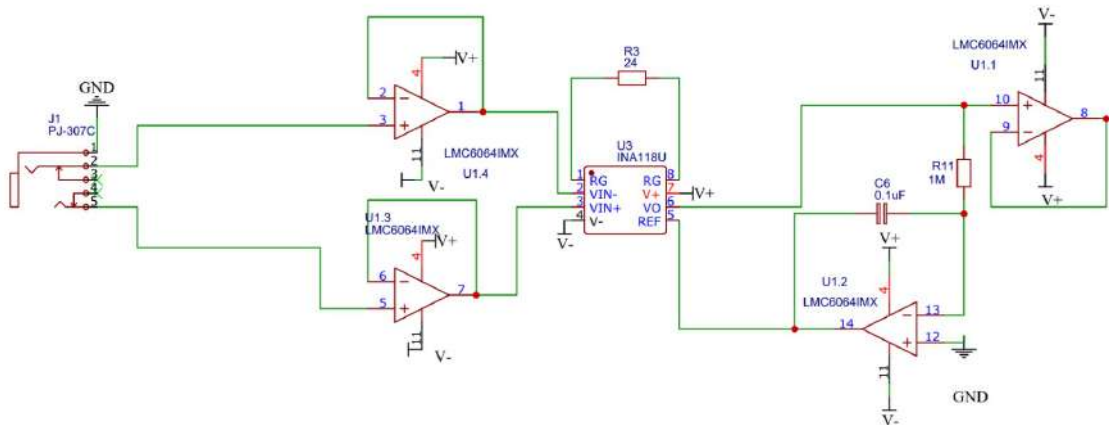


Figura 3.2: Operador de bionstrumentación.

3.1.3. Filtrado

Debido a que, las señales EMG abarcan frecuencias entre 20 Hz y 400 Hz, se diseñó un filtro pasa banda con el propósito de reducir componentes de frecuencia que se encuentren fuera de este rango.

Para el diseño de estos filtros, se optó por filtros Butterworth de segundo orden con una configuración Sallen-Key. La herramienta *Filter Design Tool* de *Webench* fue utilizada en el proceso de

diseño. La principal ventaja de los filtros Butterworth es la ausencia de ondulaciones tanto en la banda de paso como en la banda de supresión, lo que permite una transición de frecuencia suave y monótonica. Además, el uso de la topología Sallen-Key facilita la conexión en cascada de múltiples etapas sin que esto afecte la sintonización de las demás.

La señal amplificada pasa primero por un filtro pasa altas, mostrado en la figura 3.3 , el cual elimina la componente de corriente continua y las frecuencias bajas no deseadas. Este filtro se implementó para suprimir interferencias de baja frecuencia generadas por artefactos de movimiento. Posteriormente, un segundo amplificador operacional proporciona una ganancia adicional, reforzando la señal útil que mejora su resolución.

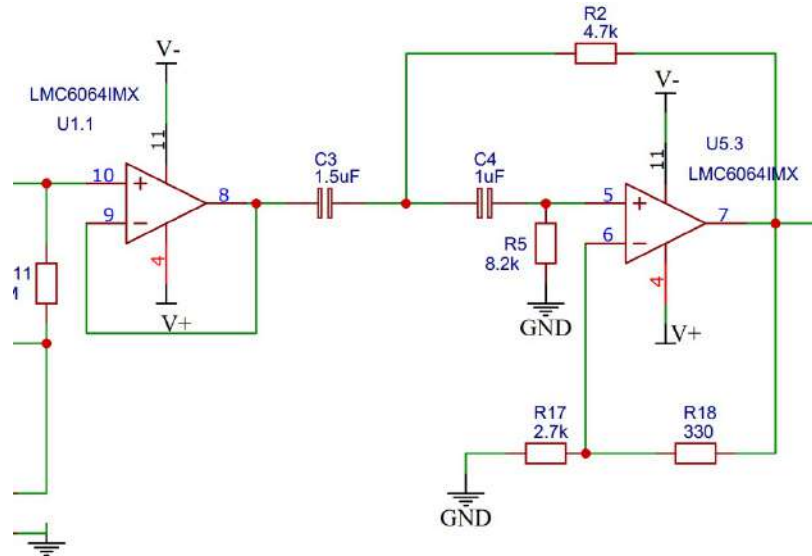


Figura 3.3: Filtro Pasa altas.

Después, la señal pasa por un filtro pasa bajas, el cual elimina las frecuencias altas no deseadas que pueden ser introducidas por el sistema de adquisición de la señal. De igual manera, se encuentra un amplificador operacional como seguidor de voltaje para proporcionar una mejor resolución. En la Figura 3.4 se observa la configuración del filtro.

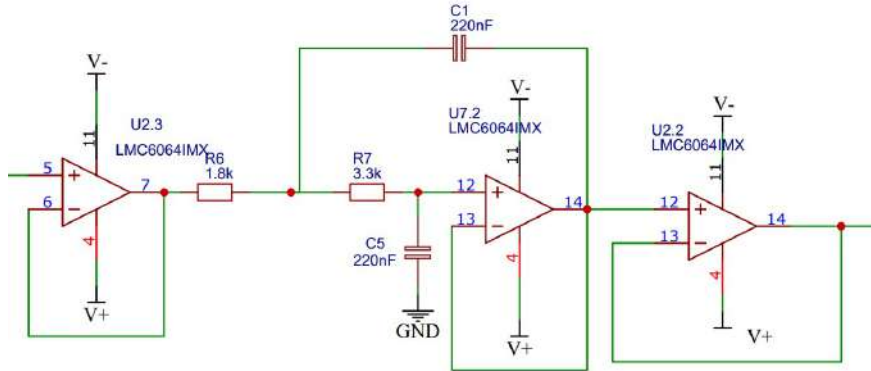


Figura 3.4: Filtro Pasa bajas.

3.1.4. Eliminación de interferencias

Para evitar distorsiones causadas por la red eléctrica, el circuito incluye un filtro Notch, que elimina específicamente la frecuencia de 60Hz, sin afectar la señal EMG. La Figura 3.5 muestra la configuración de este filtro.

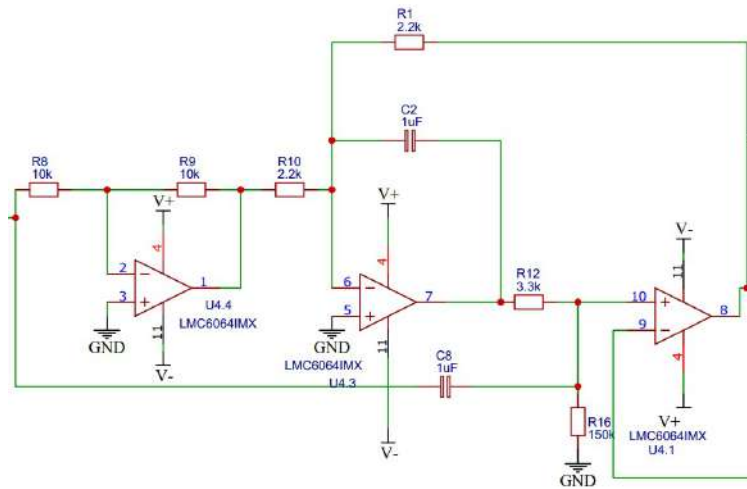


Figura 3.5: Filtro Notch.

3.1.5. Salida para digitalización

En esta etapa, la señal después de pasar por el filtro pasa bandas, se ajusta la ganancia por última vez para optimizar la amplitud de la señal antes de su salida.

Por último, se incorpora un circuito de elevación de nivel con el fin de ajustar el voltaje de salida. Esto es necesario ya que puede presentarse un voltaje de offset en el rango de milivolts en la entrada del amplificador de instrumentación.

La Figura 3.6 ilustra la configuración que se utilizará para este propósito. Finalmente, la señal acondicionada y amplificada llega al conector J1, donde se envía al sistema de adquisición de datos.

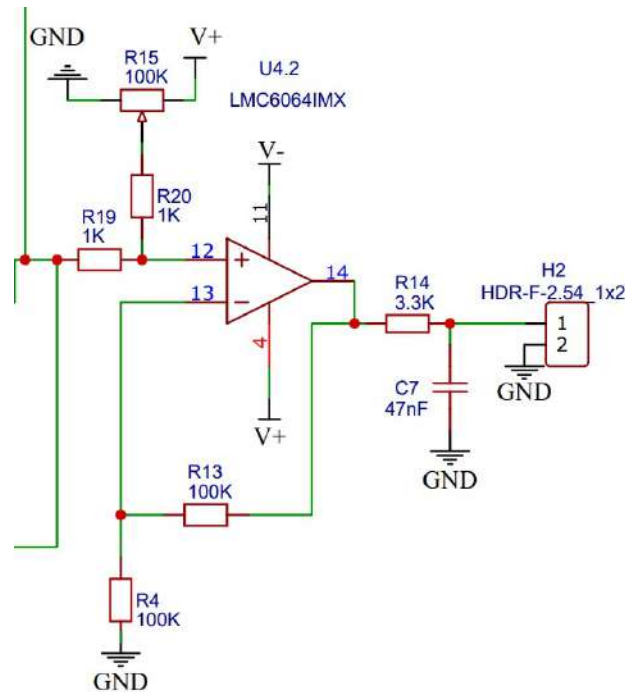


Figura 3.6: Offset y salida de la señal.

3.1.6. Manufactura de PCB

El diseño de las PCB para el sistema de adquisición se realizó en *EasyEDA* [®], utilizando montaje superficial con un empaquetado 1206 para los componentes pasivos y un empaquetado SOIC para los amplificadores operacionales, garantizando una integración compacta y eficiente [92].

3.1.7. Digitalización de la señal

La interfaz de usuario fue diseñada en *NI LabVIEW 2021* [®], e incluye tres gráficas en tiempo real para visualizar los datos de los tres canales conectados al DAQ USB-6002. Además, cuenta con opciones para seleccionar la frecuencia de muestreo, establecer la duración de la prueba, así como definir el nombre y la ubicación de almacenamiento del archivo. Para facilitar su ejecución, se generó un archivo (.exe), las señales adquiridas son digitalizadas por el sistema de adquisición USB-6002 con una resolución de 16 bits, y son enviadas a la computadora a través de un protocolo USB para su procesamiento.

3.1.8. Validación e integración del sistema

En esta etapa, se evaluó la calidad de la adquisición de las señales EMG, asegurando que los filtros implementados operaran correctamente. Para ello, se utilizó el dispositivo *Analog Discovery* como osciloscopio y generador de funciones portátil. Se generó una señal sinusoidal de 3 mV de

amplitud y frecuencia variable (0.5 Hz a 10 kHz) aplicada a la entrada diferencial de la PCB de adquisición. Se analizó la respuesta del sistema, verificando que la atenuación se realizara únicamente fuera del rango de los filtros de 20 Hz a 400 Hz y que la amplitud de salida coincidiera con la ganancia esperada de 1000, alcanzando así los 3 V.

En cuanto a la integración del sistema, una vez comprobada la comunicación entre el ejecutable y el sistema de adquisición, para brindar estabilidad y protección al sistema, se diseñó una carcasa en *SolidWorks* [®], la cual fue fabricada mediante impresión 3D en PLA. Este diseño a medida previene artefactos en la señal por movimiento de las placas que podrían comprometer la calidad de la señal o la seguridad del equipo.

En la figura 3.7 se muestra el sistema de adquisición finalizado.

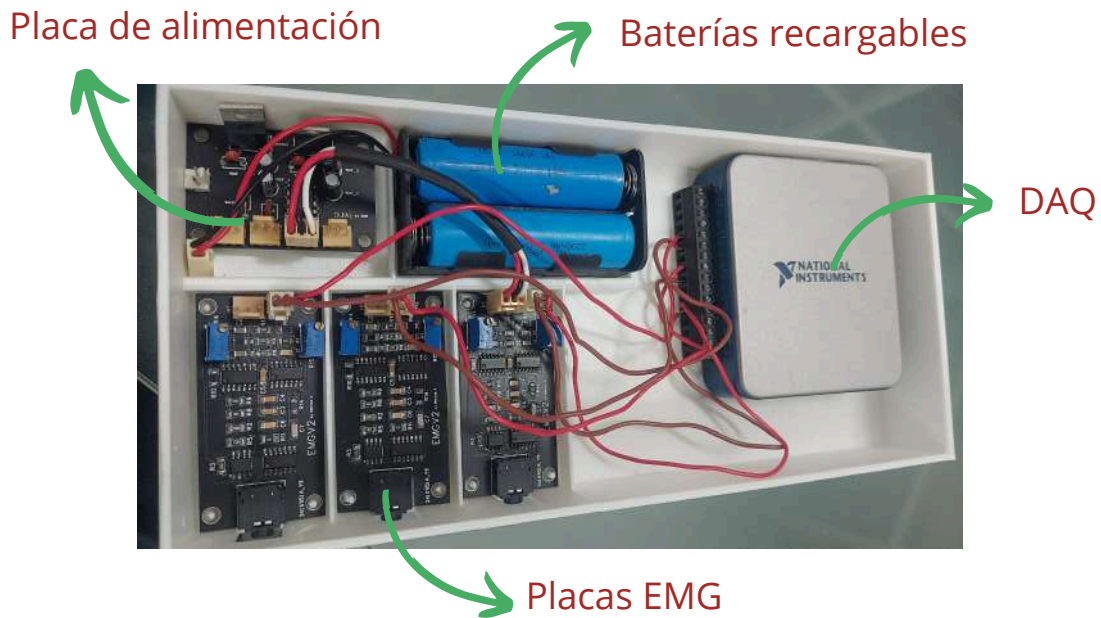


Figura 3.7: Sistema de adquisición de señales EMG compuesto por dos baterías de litio, una placa de alimentación, 3 placas EMG y el DAQ.

3.2. Base de datos

La base de datos que se presenta a continuación fue obtenida con el previo consentimiento informado de 20 participantes, 11 hombres y 9 mujeres, sanos, sin afecciones cardiovasculares, enfermedades del sistema locomotor ni del sistema nervioso, sin amputaciones ni problemas de obesidad.

Los sujetos estuvieron dentro del rango de edad de 18 a 40 años. La adquisición de las señales EMG se realizó con el objetivo de analizar cinco movimientos de la mano derecha, ejecutados por cada sujeto desde una posición inicial estandarizada.

En la figura 3.8 se observa la posición de inicio, la cual consistió en que el sujeto permaneciera sentado, con el brazo reposando sobre una superficie plana y la palma de la mano hacia arriba, sin ejercer ninguna fuerza sobre los músculos. Los movimientos solicitados fueron los siguientes:

- Flexión de todos los dedos en gesto de puño.
- Flexión del pulgar.
- Flexión del dedo índice.
- Flexión simultánea de los dedos índice, medio y anular.
- Extensión de los cinco dedos.



Figura 3.8: Posición de inicio que adoptaron los sujetos durante la toma de señales.

Cada movimiento fue realizado por las 20 personas 20 veces con fuerza sostenida durante 5 segundos por repetición, seguido de una etapa de reposo de 2 segundos para evitar la fatiga muscular prematura. El registro de la base de datos EMG para cada tipo de movimiento generó un total de 400 muestras por movimiento.

Para la adquisición de las señales EMG, se utilizaron electrodos bipolares de tipo circular, siguiendo las recomendaciones del estándar SENIAM. Estos electrodos fueron de Ag/AgCl (plata/cloruro de plata), los cuales garantizan una transición estable con un bajo nivel de ruido en la señal [93]. La elección del tamaño de los electrodos fue de 10 mm en la dirección de las fibras musculares, ya que un tamaño mayor podría perturbar la señal al detectar la actividad de otros músculos cercanos.

Los electrodos fueron colocados de acuerdo a las indicaciones anatómicas para asegurar una correcta adquisición de la señal, en la tabla 3.1 se observa su función ubicación y colocación de los electrodos.

Tabla 3.1: Músculos sensados, su ubicación y función

Nombre	Ubicación	Función
Flexor superficial de los dedos	Cara anterior del antebrazo	Responsable de la flexión de las articulaciones metacarpofalángicas e interfalángicas proximales de los dedos índice, medio, anular y meñique.
Extensor de los dedos	Cara posterior del antebrazo	Su función principal es extender las falanges proximales sobre el metacarpo.
Flexor largo del pulgar	Parte anterior del radio	Flexiona la falange distal del pulgar sobre la falange proximal y contribuye también a la flexión de la articulación metacarpofalángica del pulgar.

Antes de la colocación de los electrodos, la zona fue limpiada con alcohol y algodón [94]. Se empleó una configuración bipolar, colocando dos electrodos sobre cada músculo con una separación de 20 mm, siguiendo las recomendaciones de SENIAM [93]. Los electrodos fueron posicionados sobre el vientre muscular y alineados con la dirección de las fibras musculares, como se ilustra en la Figura 3.9. Para garantizar una colocación precisa, se realizó una palpación de la zona basada en las características anatómicas, solicitando a la persona que ejecutara los cinco movimientos definidos. Esto permitió identificar el vientre muscular mediante el tacto y posicionar los electrodos correctamente.

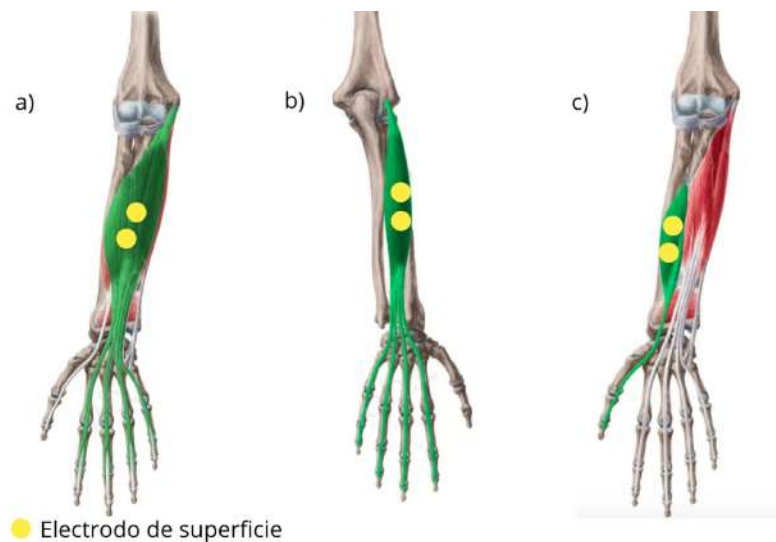


Figura 3.9: Colocación de electrodos en el músculo a) Flexor común de los dedos, b) extensor común de los dedos, y c) flexor del pulgar.

El electrodo de referencia se colocó en el hueso de la muñeca de la misma mano para garantizar la correcta adquisición diferencial de la señal de los músculos seleccionados.

El protocolo experimental consistió en que el sujeto, luego de un período inicial de relajación de 2 segundos, escuchó un sonido, emitido por el programa, el cual indicó que iniciara el movimiento con fuerza sostenida durante 5 segundos, otro sonido se emitió indicando que la persona regresara a la posición de reposo y descansara un segundo más, el tiempo de la grabación por muestra fue de 8 segundos, donde 3 fueron de reposo, con una frecuencia de muestreo de 1000 Hz. En la figura 3.10 se observa la descripción de los tiempos dedicada a la adquisición de la señal.



Figura 3.10: Protocolo de adquisición.

Entre cada repetición del mismo movimiento se dio un descanso de 2 segundos para evitar la fatiga muscular. Una vez que se completaron las 20 repeticiones, el sujeto descansó durante 60 segundos o más, de acuerdo a lo requerido, antes de proceder con el siguiente movimiento. Este proceso se repitió para cada movimiento.

Para garantizar la calidad de las señales, se le solicitó a cada persona retirarse objetos electrónicos, o metálicos, como llaves, celular y reloj, de igual manera. La laptop en la que se registraron las señales no se mantuvo enchufada a la corriente para evitar la interferencia de la señal de 60 Hz.

En la figura 3.11 se observa la manera en que la toma de señales fue realizada.

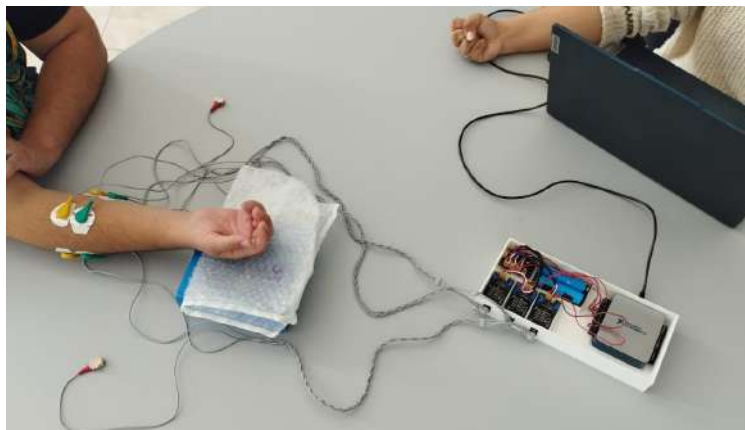


Figura 3.11: Toma de señales para la base de datos.

El sistema de adquisición de las señales EMG utilizado, junto con la metodología de colocación de los electrodos y la adquisición de los datos, fue aprobado por el comité de ética de la Facultad de Ingeniería de la Universidad Autónoma de Querétaro el 14 de marzo de 2024, con el código de aprobación CEAIFI-030-2024-TL. Todos los participantes firmaron un consentimiento informado aprobado por el comité de ética de la facultad.

3.3. Segmentación de las señales

El recorte y ventaneo de las señales fueron realizados en *Matlab 2024a*.

Las 2000 muestras obtenidas fueron organizadas por movimientos, seleccionando el segmento comprendido entre las muestras 2500 y 6999, lo que resultó en 4500 puntos de la señal original. Dado que los primeros 2 segundos y el último segundo de la grabación correspondían a periodos de reposo, estos fueron descartados para enfocarse exclusivamente en la actividad muscular asociada con el movimiento de la mano. La selección a partir de la muestra 2500, en lugar de la 2000, se realizó para considerar posibles variaciones en el tiempo de reacción del participante ante el estímulo auditivo.

Se experimentó con distintos niveles de superposición en la segmentación de las señales EMG, evaluando traslapes del 70 %, 75 % y 80 %. Para un traslape del 70 %, cada ventana se desplazó 60 ms, obteniendo 72 ventanas por señal. Con 75 %, el desplazamiento fue de 50 ms, con 87 ventanas por señal. Finalmente, con un traslape del 80 %, el desplazamiento se redujo a 40 ms, generando 113 ventanas.

Estos valores experimentales de porcentaje de traslape, se basan en la capacidad de las ventanas superpuestas para capturar de manera continua la información durante el análisis de las señales, lo cual es esencial para los sistemas de clasificación en tiempo real [95, 55].

De acuerdo a [18], la segmentación en ventanas de corta duración es adecuada para estimar movimientos musculares, dado que este lapso es suficiente para capturar los patrones relevantes de la señal EMG asociados a un movimiento específico. El uso de ventanas con superposición incrementa la densidad de la información, ya que genera más segmentos de datos a partir del mismo conjunto de señales. Esta mayor densidad proporciona una mayor cantidad de ejemplos para el entrenamiento y la validación del modelo de aprendizaje automático, lo que puede mejorar la capacidad de generalización del modelo y reducir el riesgo de sobreajuste [95, 33].

Sustituyendo la ecuación 2.1, se obtuvo la ecuación 3.1, que da como resultado 87 ventanas.

$$N = \left\lfloor \frac{4500 - 200}{200 - 150} \right\rfloor + 1 = \left\lfloor \frac{4300}{50} \right\rfloor + 1 = \lfloor 86 \rfloor + 1 = 87 \quad (3.1)$$

Teniendo en total una matriz de 174,000 muestras por 3 sensores por 200 puntos en el tiempo.

3.4. Diseño del sistema de clasificación

La base de datos creada para la clasificación, se dividió en dos grupos: uno para entrenamiento y otro para prueba. El 20 % de los datos se usa para prueba y el 80 % para entrenamiento. Para garantizar que la separación de los datos sea reproducible en diferentes ejecuciones, se utilizó una semilla aleatoria en la función de división de datos, lo que permite obtener los mismos subconjuntos en cada ejecución del código [96].

3.4.1. Codificador y Decodificador

En *Python* se desarrolló el codificador y decodificador de las señales con el fin de obtener el vector latente que predice la señales para el proceso de clasificación.

Inicialmente, se diseñó un modelo de autoencoder alimentado con un vector que contenía los datos de los tres sensores simultáneamente. Posteriormente, este se comparó con tres modelos de autoencoder adicionales, cada uno entrenado con los datos de un solo sensor por separado. De esta manera, se evaluaron dos enfoques: uno en el que la señal combinada de los tres sensores se procesaba en un solo modelo, y otro en el que se usaban tres modelos independientes para reconstruir la señal de cada sensor por separado.

Para tener características relevantes de las señales con el fin de reconstruirlas, se utilizó una arquitectura basada en redes neuronales recurrentes, que incluye capas LSTM y GRU.

Para medir la diferencia entre las entradas originales y las entradas reconstruidas se probaron diferentes funciones de pérdida como MSE, RMSE, MEA Y MAPE. Distintos optimizadores y tasas de aprendizaje fueron utilizadas para el ajuste de parámetros en el modelo como el descenso por gradiente y ADAM.

La reconstrucción de las señales fue el objetivo de este enfoque, ya que, se buscó encontrar una representación significativa de la señal original para utilizarla en la clasificación.

Una vez obtenido el modelo de codificación y decodificación, se extrajo la capa latente del encoder del autoencoder. El encoder aprendió una representación comprimida de las señales EMG, lo que permite aprovechar el conocimiento adquirido sobre la estructura de las señales sin tener que entrenar desde cero en nuevas tareas.

El objetivo de extraer esta capa latente es utilizar la representación aprendida por el encoder en la clasificación. Para garantizar que el conocimiento previamente adquirido no se vea alterado, se congelaron las capas del encoder. De esta forma, sus pesos no se actualizarán durante el entrenamiento de los nuevos modelos, manteniendo la eficiencia y precisión de las representaciones latentes.

Los modelos nuevos se construyeron utilizando la capa latente como salida final, en lugar de las señales originales. Con estos nuevos modelos, se predijeron las señales tanto del conjunto de entrenamiento, con el que se entrenó la red neuronal, como del conjunto de test, con el que se evaluó el desempeño del modelo.

3.4.2. Clasificador EMG

Para el clasificador, se desarrollaron dos modelos de red neuronal profunda con arquitectura densa para la clasificación multiclase de señales para su comparación. La entrada del modelo fue un vector unidimensional correspondiente a los datos EMG de tres sensores. Antes de alimentar los datos al modelo, se aplicó una capa de normalización, lo cual ayuda a mejorar la estabilidad y eficiencia durante el entrenamiento al reducir la varianza entre las características.

En el primer modelo de clasificador, se realizó una partición basada en los sujetos de la base de datos. Los datos de 16 personas al azar se asignaron al conjunto de entrenamiento, los de 2 personas a validación y las últimas 2 personas para prueba.

En el segundo modelo se realizó una partición aleatoria de todos los datos, asignando el 80 % para entrenamiento y el 20 % para prueba. Dentro del conjunto de entrenamiento, el 10 % se reservó para validación.

Para el diseño del clasificador con una red neuronal profunda densa, se realizaron primero una

serie de pruebas exploratorias con distintas configuraciones de arquitectura, activaciones, normalización y parámetros de entrenamiento. Inicialmente se construyó una red con 5 capas ocultas, evaluando su desempeño, sin mostrar una mejora en la precisión. Posteriormente se amplió a una arquitectura con 10 capas ocultas, siguiendo un esquema de reducción progresiva del número de neuronas en cada capa con un decremento aproximado del 50 %, lo que permitió un mejor ajuste a la complejidad de la clasificación. Finalmente, se optó por una configuración de 8 capas ocultas con las siguientes dimensiones: 2200, 1100, 500, 250, 125, 75, 30 y 15 neuronas, la cual ofreció un equilibrio adecuado entre precisión y tiempo de entrenamiento.

Durante este proceso también se evaluaron distintas funciones de activación: ReLU, sigmoid, tangente hiperbólica, GELU y ELU.

Para evitar el sobreajuste, se implementaron diferentes estrategias. Se probó el modelo con y sin capa de normalización de entrada, siendo más estable y rápida la convergencia con esta capa activada. También se evaluó el efecto de incluir capas de Dropout con tasas del 10 %, 20 % y 30 % después de cada capa oculta.

En la capa de salida se utilizó una capa densa con 5 neuronas, correspondientes a las cinco clases de movimientos. Para esta capa se eligió la función de activación softmax, adecuada para problemas de clasificación multiclase, ya que genera una distribución de probabilidad sobre todas las clases.

En cuanto al algoritmo de optimización, se evaluaron distintos optimizadores: SGD, RMSprop y Adam. De igual manera, se probaron diferentes tasas de aprendizaje en un intervalo desde 0.1 hasta 0.00001.

Finalmente, el tamaño del lote también fue ajustado experimentalmente. Se probaron tamaños de 512, 256, 128, 64 y 32.

El modelo fue entrenado utilizando un 10 % del conjunto de entrenamiento como validación interna. Se implementaron dos estrategias para evitar el sobreentrenamiento: *EarlyStopping*, que detenía el entrenamiento si la métrica de validación no mejoraba durante 30 épocas, y *ModelCheckpoint*, que guardaba automáticamente la mejor versión del modelo.

3.5. Interfaz Gráfica

Para desarrollar la interfaz gráfica, se definieron las clases de movimiento en la clasificación. Cada movimiento está representado por un número específico, lo que permite que el modelo pueda interpretar y dar resultados. De esta manera, cuando se haga una predicción, se podrá traducir el resultado a un nombre de movimiento.

Posteriormente, se estableció un mecanismo para recibir la entrada del usuario. Se diseñó un cuadro de texto en el que la persona debe ingresar un número que representa una señal específica dentro del conjunto de datos. Este número debe estar dentro de la cantidad de datos que se tienen del conjunto de test, por lo que el programa verifica que el valor ingresado sea válido. Si el usuario introduce un número fuera del rango o un dato no numérico, se le muestra un mensaje de error solicitando una corrección.

Antes de generar un nuevo gráfico, se elimina cualquier gráfico anterior. Esto evita que se acumulen múltiples visualizaciones en la interfaz o se reescriban datos.

Al realizar la predicción del movimiento correspondiente a la señal ingresada, esta se procesa con el modelo de clasificación obtenido. Este modelo analiza los datos y devuelve un resultado que indica el tipo de movimiento que ha logrado identificar. Simultáneamente, se obtiene la etiqueta

real de la señal desde el conjunto de datos y se compara con la predicción del modelo. Si ambas coinciden, se considera que la predicción es correcta; de lo contrario, se indica que el modelo ha cometido un error en la clasificación.

Después de obtener la predicción, se generan gráficos para visualizar la señal real y la señal predicha. Dado que se utilizan tres sensores, se presentan tres gráficos independientes. En donde el primero corresponde al flexor común de los dedos, el segundo al flexor del dedo pulgar y el tercero al extensor común de los dedos. En cada gráfico se trazan dos curvas, una azul que representa la señal real de la base de datos y otra roja que muestra la señal obtenida por el modelo de predicción.

La interfaz permite que el usuario interactúe con el usuario al introducir un nuevo número de señal, se actualizan los gráficos y se muestran los nuevos resultados. Este proceso se puede repetir cuantas veces se requiera para analizar diferentes señales y evaluar el desempeño del modelo de predicción y clasificación.

3.6. Validación del modelo

Finalmente, cada uno de los modelos de autoencoder fueron evaluados por medio del valor de pérdida que presentaban en la evaluación y gráficamente comparando la señal real con la predicha,

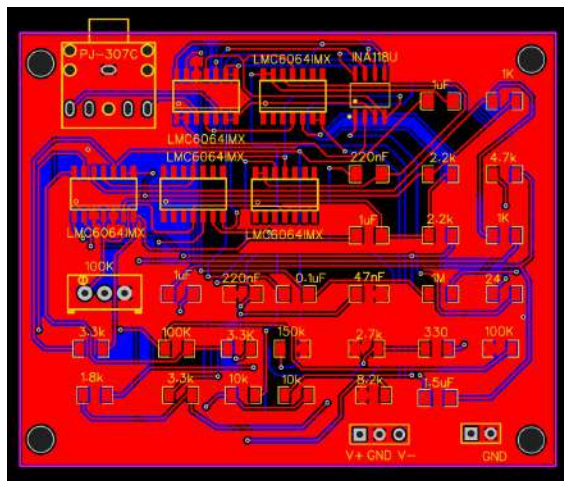
Para los modelos del clasificador se evaluaron las siguientes métricas: precisión, sensibilidad, especificidad y eficiencia. Por ultimo, se seleccionó el modelo que presentó las mejores características con base a estas métricas.

Resultados

En esta sección se presenta el análisis de los resultados de la metodología.

4.1. Sistema de adquisición

Se diseñó la PCB como se muestra en la figura 4.1 en el software *EasyEda*.



(a) Diseño PCB de adquisición.



(b) Modelo 3D de la PCB de adquisición.

Figura 4.1: Manufactura de la PCB de alimentación.

La carcasa fue diseñada en *SolidWorks* (®), como se muestra en la figura 4.2, para evitar daños en el sistema de adquisición y artefactos por movimiento del circuito.



Figura 4.2: Sistema de adquisición EMG

4.2. Segmentación y ventaneo

Se obtuvieron 4,500 muestras en el tiempo, por los 20 pacientes, por las 20 repeticiones de cada movimiento. Sin embargo, para su procesamiento es indispensable segmentar y ventanear para realizar un análisis preciso y disminuir la carga computacional.

Las señales EMG provenientes de los tres sensores, una vez recortadas los segundos que se dieron de reposo, fueron segmentadas en 200 muestras. Esta segmentación permitió analizar con mayor detalle la evolución de la señal a lo largo del tiempo y facilitar su uso en el entrenamiento de los modelos con redes neuronales.

En la figura 4.3 se muestra la comparación de la señal de 4500 puntos contra la segmentada de 200 puntos de los 5 movimientos, para los tres sensores, en donde el eje de las ordenadas corresponde al voltaje generado por la señal y el de las abscisas a los puntos en el tiempo.

En la figura 4.3a se observa el primer sensor corresponde al músculo flexor común de los dedos, donde se observa que el voltaje generado alcanza hasta 0.2 V, el sensor del flexor del pulgar hasta 0.5 V y el del extensor 0.4 V. En particular, el sensor 1 muestra una señal de menor amplitud, mientras que el sensor 2 registra valores más altos, lo que indica una mayor activación del flexor del pulgar en comparación con los otros músculos. El sensor 3 presenta un comportamiento intermedio, con una variabilidad que sugiere la presencia de activaciones musculares de diferente intensidad.

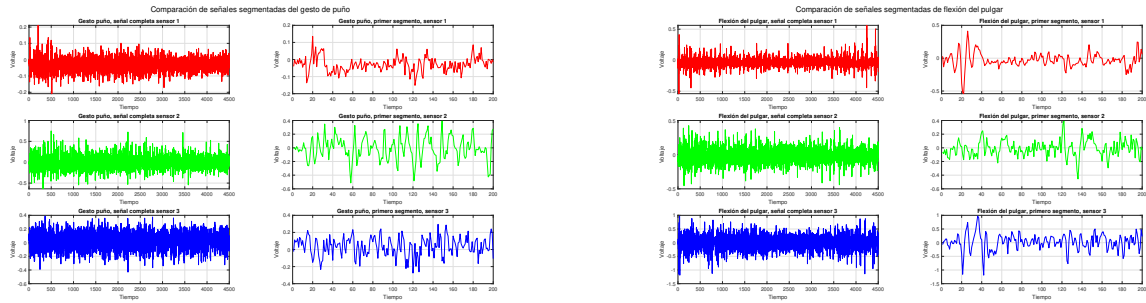
La figura 4.3b corresponde a la flexión del pulgar, en la cual se observa una mayor actividad eléctrica en el segundo sensor, con voltajes de hasta 0.5, correspondiente al músculo flexor del pulgar, en comparación con el sensor 1, el músculo flexor común. El sensor 3, alcanza picos de voltaje de hasta 1 v para este movimiento, pero se sostiene en 0.5 v.

Las gráficas de la señal completa y primer segmento de la acción de reposo se observan en la figura 4.3c. En estas imágenes los tres sensores tienen voltajes muy bajos, ya que ningún músculo está siendo activado. La señal del músculo flexor común de los dedos, se encuentra sobre 0 v, la del flexor del pulgar llega hasta 0.1 v, siendo el sensor 3, el extensor de los dedos con mayor voltaje registrado superior a los 0.5 v.

En la figura 4.3d se muestran las gráficas de la extensión de los dedos de la mano, al ser el músculo que se activa para este movimiento, el extensor de los dedos se presenta con mayor con

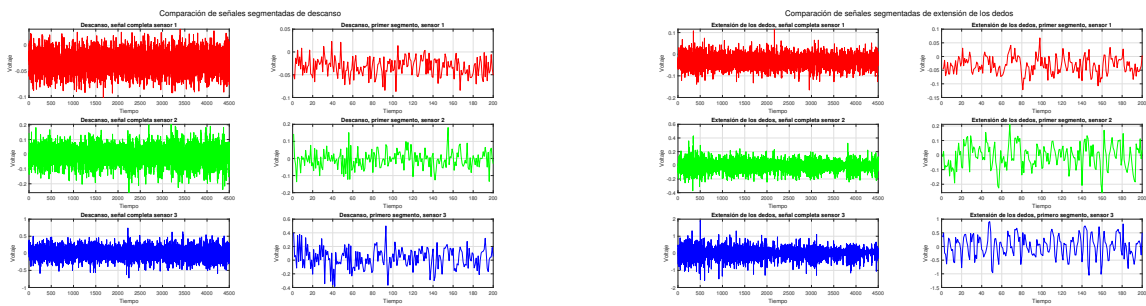
picos de voltaje de hasta 2 v, con voltaje constante de 1 v. Mientras que los primeros dos sensores tienen poca actividad con 0.1 para el del flexor común y de hasta 0.4 para el flexor del pulgar.

Para el último movimiento, el flexor común de los dedos, sus gráficos se muestran en la figura 4.3e. En el primer sensor se observan una gran cantidad de picos de voltaje de 0.5 v, en el sensor del flexor del pulgar se obtiene un voltaje constante de 0.2 v, mientras que para el sensor del músculo extensor se tienen picos de hasta 0.4 v.



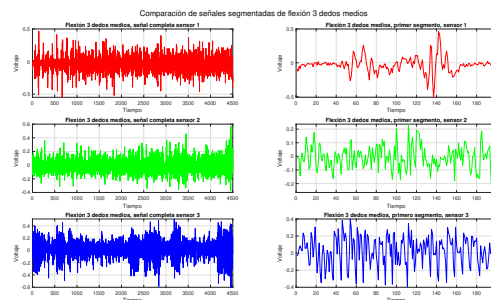
(a) Puño

(b) Flexión del pulgar



(c) Descanso

(d) Extensión de los dedos



(e) Flexión de tres dedos medios

Figura 4.3: Segmentación de las señales de los diferentes movimientos de la mano.

En las gráficas obtenidas de los movimientos, se aprecia que la gráfica de la señal completa de 4500 puntos está saturada de información, lo que dificulta distinguir claramente los voltajes de la señal a lo largo del tiempo. En contraste, al analizar la señal segmentada de 200 puntos, se aprecia

con mayor claridad la forma de onda y los patrones de activación muscular de cada sensor.

Se experimentó con diferentes porcentajes de superposición de las ventanas: inicialmente con un 70 %, posteriormente se probó con un 75 %, lo que resultó en una mejora en la precisión. Al aumentar la superposición al 80 %, la mejora en la precisión fue mínima, pero el tiempo de procesamiento se incrementó notablemente, lo que sugiere que un 75 % de superposición es un buen compromiso entre precisión y eficiencia. Por lo que para mejorar la representatividad del conjunto de datos, las señales fueron ventaneadas con un traslape de 75 %.

El ventaneo de las señales de los tres sensores se puede observar en la figura 4.4.

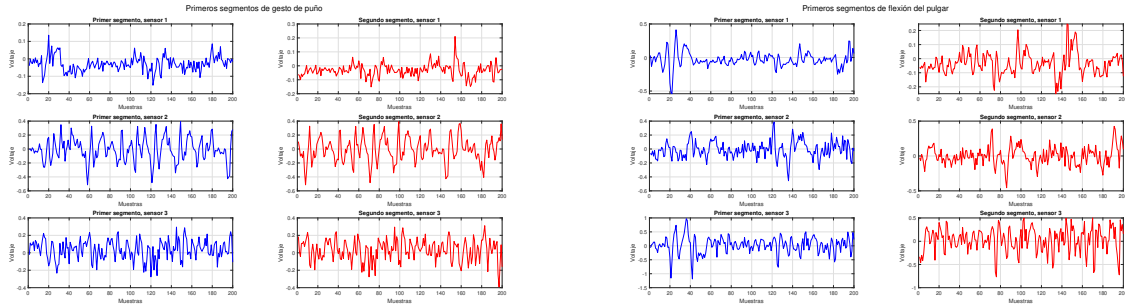
La figura 4.4a muestra la comparación de la primera y segunda ventana del gesto de puño. Donde la señal en azul representa el primer segmento de la muestra 1 a la 200, mientras que la señal en rojo corresponde a la segunda ventana, que comienza en la muestra 51 y termina en la 250. Se nota una alta correlación entre ambas ventanas, lo que indica que la superposición del 75 % permite conservar la continuidad de la señal y evitar la pérdida de información relevante.

Los segmentos de flexión del pulgar se observan en la figura 4.4b, se observa que ambos segmentos preservan la secuencia de los datos ya que, el inicio de la señal en rojo se observa en la muestra 50 de la señal en azul.

La figura 4.4c representa la acción de reposo. Aquí, la señal tiene menor variabilidad y amplitud en comparación con los movimientos activos. La superposición sigue manteniendo la continuidad de la señal, pero se observa que los cambios entre ventanas son menos notorios debido a la menor activación de los músculos.

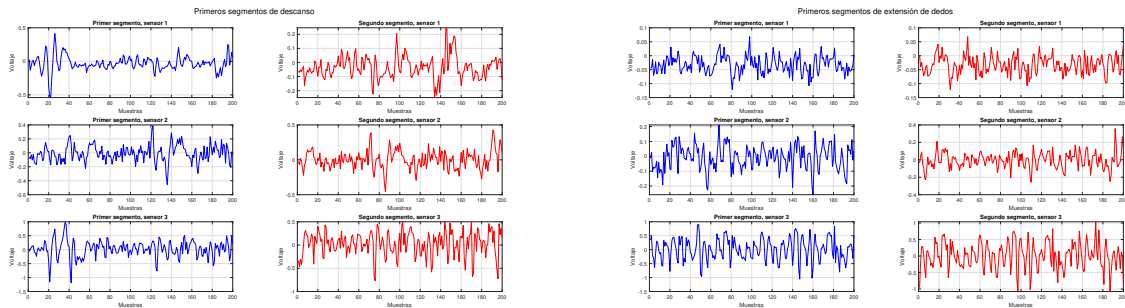
Para la extensión de los dedos de la mano, las gráficas se observan en la figura 4.4d. Las ventanas consecutivas conservan la estructura de la señal, permitiendo una buena continuidad. Este movimiento muestra una activación mayor en el sensor del músculo extensor, con una menor amplitud de señal en los músculos flexores por la naturaleza del movimiento.

Finalmente, la Figura 4.4e presenta la comparación de ventanas para la flexión de los tres dedos medios. La superposición sigue mostrando su eficacia al mantener la coherencia entre segmentos consecutivos, asegurando que la información no se pierda entre ventanas.



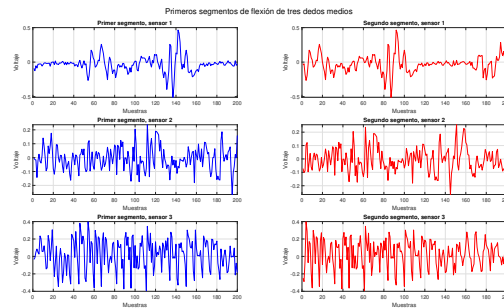
(a) Movimiento puño.

(b) Flexión del pulgar.



(c) Descanso.

(d) Extensión de los dedos de la mano.



(e) Flexión de los tres dedos medios.

Figura 4.4: Comparación de la ventana 1 y la ventana 2 para los 5 movimientos.

Finalmente, la base de datos contiene 174,000 muestras, de la cual 34,800 señales con 200 puntos en el tiempo son del gesto puño, flexión del pulgar, descanso, extensión de la mano y flexión de tres dedos, en orden.

La segmentación y ventaneo de las señales EMG permitió obtener una mayor cantidad de datos a partir del mismo conjunto de señales. Este enfoque resultó en una mayor densidad de información, lo que mejoró la representatividad de los datos utilizados para el entrenamiento y la validación del modelo de aprendizaje automático.

4.3. Codificador y decodificador

Se realizó una partición aleatoria de todos los datos, asignando el 80 % para entrenamiento y el 20 % para prueba. Con el conjunto de entrenamiento se alimentaron los modelos de autoencoder.

El primer modelo de autoencoder integraba las señales de los tres sensores EMG, utilizando una arquitectura secuencial de capas GRU y LSTM. Esta red incluía dos capas LSTM de 500 y 400 unidades respectivamente, seguidas por dos capas GRU de 350 y 200 unidades con función de activación RELU. El vector latente generado pasaba por una capa densa de 2000 neuronas con activación Sigmoid. Para la reconstrucción (decoder), se usaron tres capas recurrentes: primero una GRU con 500 unidades, seguida de una LSTM de 300 unidades, y una GRU con 200 neuronas, finalizando con una capa para generar la salida reconstruida por cada sensor.

Al observar que el modelo no era capaz de predecir las señales de los tres sensores, se optó por un modelo para cada sensor. Este enfoque permite que el modelo se especialice en las características propias de cada sensor, lo cual resultó en una mejor reconstrucción y una pérdida más baja en comparación con el modelo general.

La arquitectura de cada modelo comienza con una capa de entrada que recibe las señales de 200 puntos. Para facilitar el aprendizaje y estabilizar el entrenamiento, se aplicó una capa de normalización sobre la señal de entrada, asegurando que los valores tuvieran media cero y desviación estándar uno.

En el encoder, se empleó primero una capa LSTM con 200 unidades, utilizando la función de activación ELU y configurada para retornar secuencias. Posteriormente, se agregó una capa GRU con 2000 unidades, encargada de comprimir toda la secuencia en un vector latente.

El decoder comienza con una capa que repite el vector latente permitiendo la reconstrucción de la señal. Luego, se utilizaron dos capas recurrentes: una LSTM de 400 unidades y una GRU con 300 unidades, ambas con activación ELU y configuradas para retornar secuencias. Finalmente, se integró una última capa LSTM con 200 unidades para generar la salida reconstruida, manteniendo la forma original de la señal de entrada.

Para el entrenamiento, se usó la función de pérdida error cuadrático medio (MSE), adecuada para tareas de reconstrucción. El optimizador seleccionado fue Adam, con una tasa de aprendizaje de 0.001.

El modelo fue entrenado por un máximo de 200 épocas, con un tamaño de lote de 128 muestras, y utilizando un 10 % de los datos para validación.

A lo largo del diseño, se exploraron diferentes configuraciones. Primero, se probó con arquitecturas de cinco capas, que luego se aumentaron a ocho, sin encontrar mejoras significativas en la pérdida de reconstrucción. Por tanto, se optó por una arquitectura optimizada de siete capas (3 encoder, 4 decoder). Asimismo, se observó que cuando las neuronas eran escasas (16) o excesivas (500), no lograban representar eficientemente los 200 puntos de cada señal; fue a partir de configuraciones intermedias (200 a 400 neuronas) donde se alcanzaron mejores resultados.

Una vez definidas las capas y neuronas, se evaluaron las funciones de activación, se evaluaron ReLU, GELU y ELU. Aunque ReLU y GELU presentaban buen desempeño para valores positivos, la activación ELU fue la que permitió capturar con mayor fidelidad la morfología real de la señal, ya que admite valores negativos. Por esta razón, ELU se implementó como función de activación principal en el modelo por sensor.

El optimizador Adam, con una tasa de aprendizaje de 0.001, ofreció mejores resultados frente a otras opciones como Nadam y SGD. Además, se probó con distintos tamaños de lote: 32, 64, 128 y 256. Finalmente, se eligió un batch size de 128 o 256, ya que equilibraba el tiempo de entrenamiento

y el consumo de memoria sin afectar negativamente el rendimiento.

Al comparar los resultados entre el modelo unificado y los modelos individuales por sensor, estos últimos demostraron ser más eficaces. Cada modelo individual constó de una capa LSTM de 200 unidades con activación ELU, seguida de una capa GRU de 2000 unidades en el encoder.

Este proceso de optimización de hiperparámetros fue repetido para ambos enfoques. Los callbacks utilizados incluyeron EarlyStopping con una paciencia de 120 épocas en cada uno, y Model-Checkpoint para guardar los mejores modelos durante el entrenamiento. Esta estrategia permitió obtener representaciones latentes que mejoraron la reconstrucción y optimizaron el rendimiento del clasificador final basado en dichas representaciones.

Una ventaja del enfoque por separado, fue la optimización computacional. Al entrenar los modelos de manera independiente, se ajustó el tamaño de los lotes para cada sensor, lo que permitió un mejor aprovechamiento de los recursos computacionales sin comprometer la calidad del entrenamiento. Para el sensor 2 se aumentó el tamaño del lote a 256, mientras que para los otros sensores se utilizó un lote de 128.

En las figuras 4.5, 4.6 y 4.7 se muestra la predicción de las señales para los tres sensores obtenidas del autoencoder único.

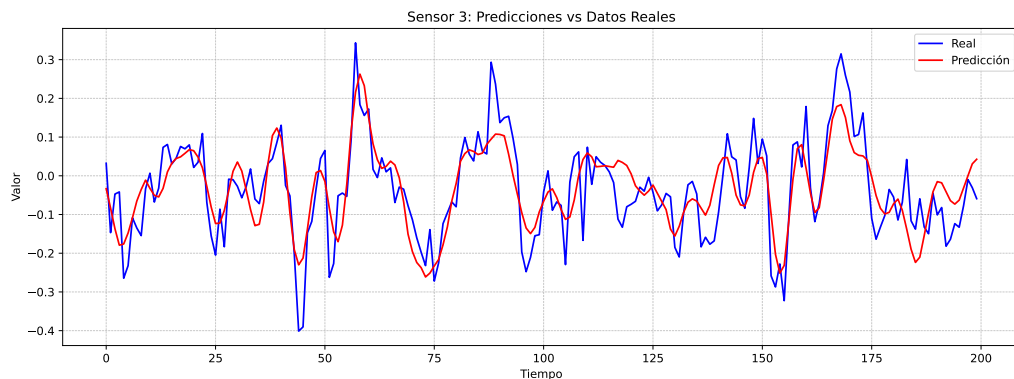


Figura 4.5: Predicción y señal real del sensor 1 del autoencoder global.

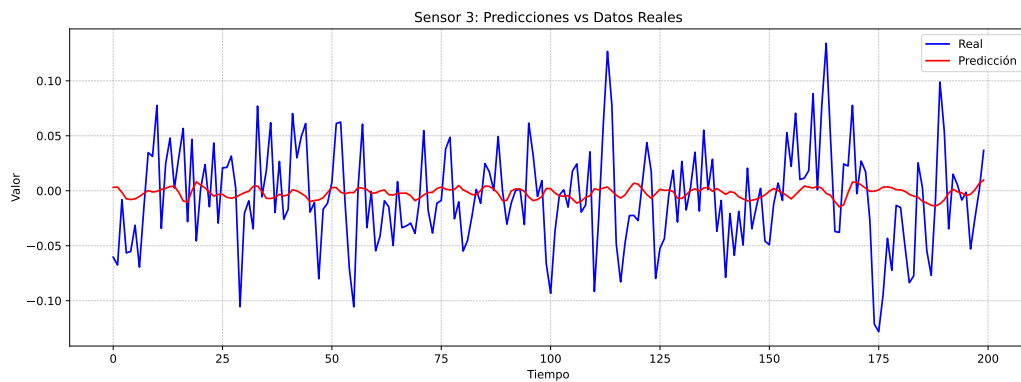


Figura 4.6: Predicción y señal real del sensor 2 del autoencoder global.

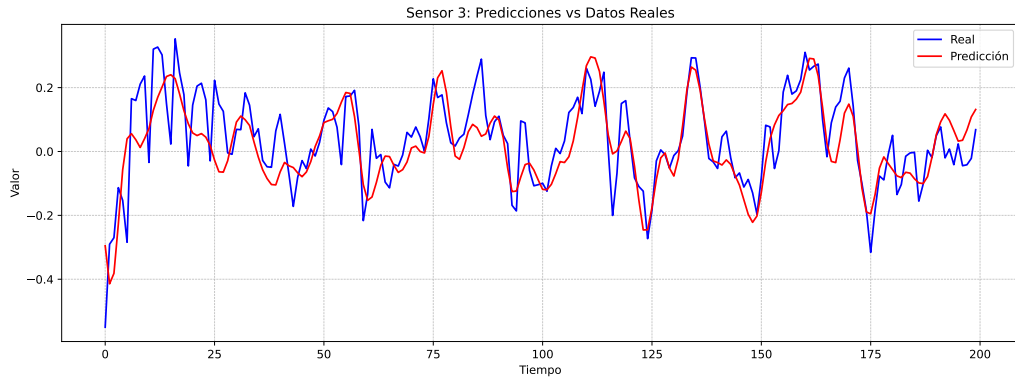


Figura 4.7: Predicción y señal real del sensor 3 del autoencoder global.

En las figuras 4.8, 4.9 y 4.10, se observan la predicción de los datos obtenidos del autoencoder individual para el sensor 1, sensor 2 y sensor 3 respectivamente. Donde la línea azul es la señal real, mientras que la roja es la predicha.

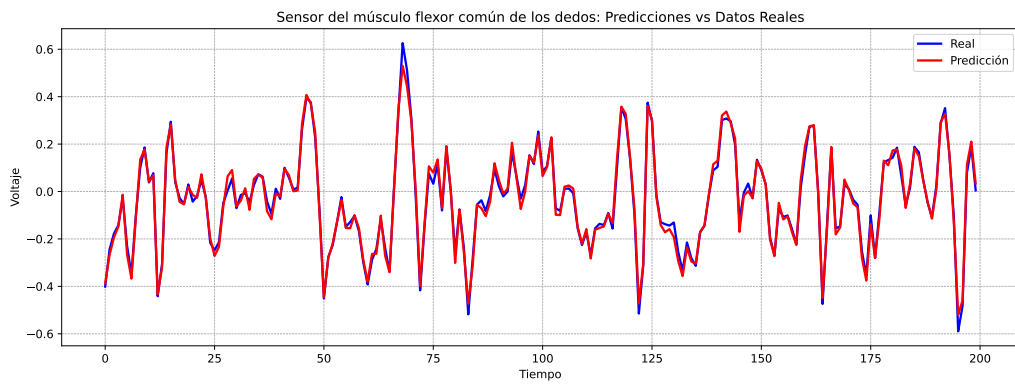


Figura 4.8: Predicción y señal real del sensor 1 del autoencoder individual.

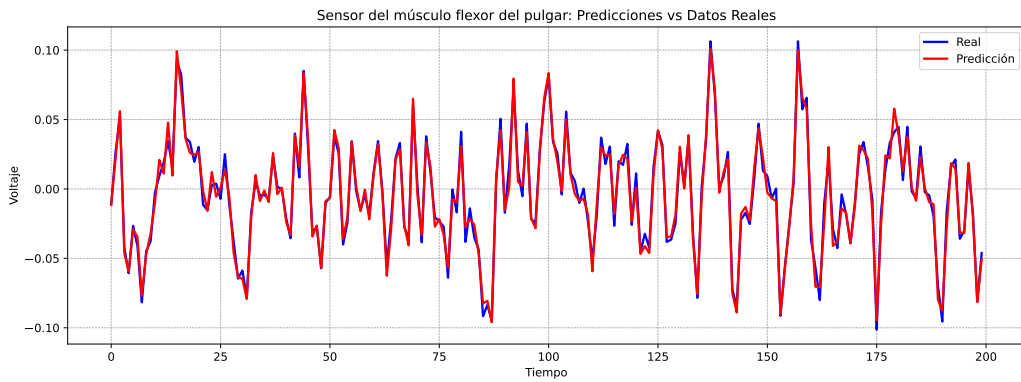


Figura 4.9: Predicción y señal real del sensor 2 del autoencoder individual.

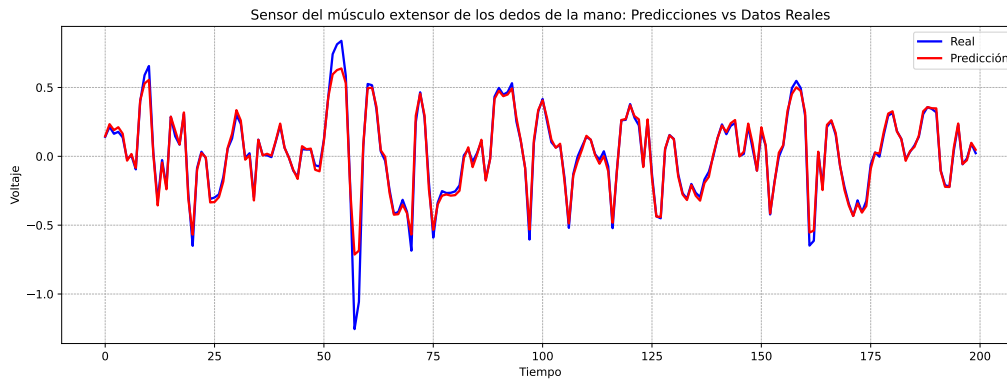


Figura 4.10: Predicción y señal real del sensor 3 del autoencoder individual.

Se muestra una mejor eficacia en la predicción de señales del autoencoder particular para cada sensor. Lo cual indica que las características extraídas en el encoder son de utilidad.

Tras obtener el modelo de codificación y decodificación, se extrajo la capa latente del codificador para construir nuevos modelos basados en estas representaciones comprimidas de las señales. Para garantizar que la información aprendida en las capas latentes no se viera afectada durante el entrenamiento del clasificador, se congeló el proceso de actualización de pesos en estas capas. De esta manera, las representaciones latentes se mantuvieron intactas, asegurando que los modelos preservaran las características más relevantes de las señales originales.

Con los nuevos modelos, se generaron predicciones tanto en el conjunto de entrenamiento (del cual, posteriormente el 10 % se divide para validación) como en el conjunto de test, los cuales fueron concatenados para entrenar, validar y evaluar con un solo vector el clasificador.

4.4. Clasificador EMG

Se realizó una partición basada en los sujetos de la base de datos. Los datos de 16 personas al azar se asignaron al conjunto de entrenamiento, los de 3 personas a validación y las últimas 3 personas para prueba. Durante el entrenamiento y la validación, el modelo logró una precisión del 92 %; sin embargo, al evaluar el conjunto de prueba, la precisión se redujo significativamente al 60 %. Esto demostró que el clasificador alimentado con esta configuración de base de datos, no generalizaba bien a sujetos no vistos previamente.

Por ello, en un segundo experimento, se realizó una partición aleatoria de todos los datos, asignando el 80 % para entrenamiento y el 20 % para prueba. Dentro del conjunto de entrenamiento, el 10 % se reservó para validación. Con esta partición de la base de datos se obtuvo una precisión de clasificación en el conjunto de entrenamiento de 96.5 %, en validación alcanzando hasta 88.98 % y en el conjunto de evaluación de 88.81 %.

Una vez obtenidos los conjuntos de entrenamiento, validación y test, se diseñó y entrenó una red neuronal profunda para la clasificación de señales, en el cual se aplicó normalización de los datos de entrada, lo que facilitó la convergencia del modelo al reducir la variabilidad entre las características.

En la tabla 4.1 se detallan los hiperparámetros utilizados para el entrenamiento del modelo de clasificación propuesto.

Variable	Descripción
Capas ocultas	8 capas densas con un número decreciente de neuronas: 2200, 1100, 500, 250, 125, 75, 30 y 20
Capa de salida	5 neuronas, correspondiente a las 5 clases de movimientos
Activación oculta	ELU, mejora estabilidad del entrenamiento y evita neuronas muertas
Activación de salida	Softmax, adecuada para clasificación multiclase
Dropout	20 % después de cada capa oculta, para prevenir sobreajuste
Tasa de aprendizaje	0.00003, elegida tras probar valores entre 0.1 y 0.00001
Optimizador	Adam, el más estable y preciso
Batch size	32, fue el mínimo valor que no comprometió el rendimiento computacional
Épocas	500 máximas.

Tabla 4.1: Resumen de la arquitectura y parámetros del clasificador

En la figura 4.11 se observa la arquitectura del clasificador para 5 movimientos de la mano.

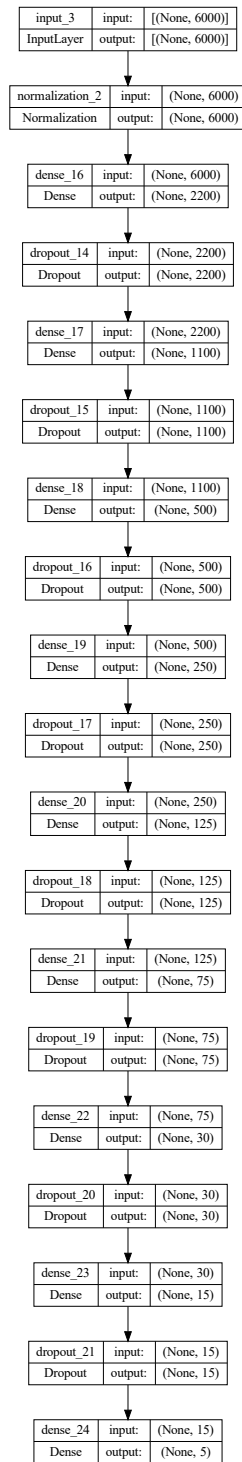


Figura 4.11: Representación del modelo de clasificación.

En la Figura 4.12 se muestra la evolución de la precisión tanto en el conjunto de entrenamiento como en el de validación. Al inicio del proceso de entrenamiento, ambas curvas aumentan rápidamente, lo que indica que la red neuronal logra aprender patrones relevantes desde las primeras épocas. Posteriormente, la precisión en validación se estabiliza alrededor de un valor constante, cerca del 89% considerando todas las clases y muestras, mientras que la precisión en entrenamiento continúa incrementando hasta alcanzar valores cercanos al 96%. Esta diferencia sugiere que el modelo comienza a ajustarse más a los datos de entrenamiento que a los de validación. Dado que ambas curvas se mantienen cerca, se concluye que el modelo presenta la capacidad de clasificación.

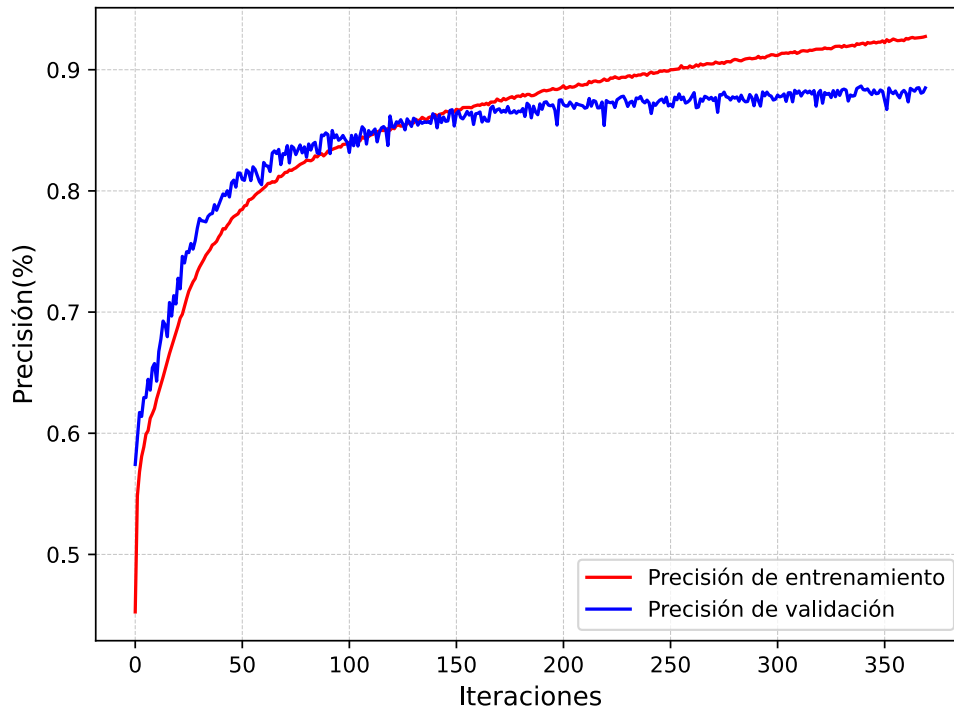


Figura 4.12: Gráfica de la precisión en entrenamiento y validación.

Por otro lado, en la Figura 4.13 se representa la pérdida o el error en función de las épocas para los mismos conjuntos. La pérdida de ambas curvas disminuye durante las primeras épocas, lo que refleja un aprendizaje eficiente. No obstante, a partir de aproximadamente la época 100, la pérdida en validación comienza a estabilizarse, mientras que la pérdida en entrenamiento sigue descendiendo de manera continua. Después de la época 300, se observa un ligero incremento en la pérdida de validación, lo cual refuerza la presencia de un ligero sobreajuste.

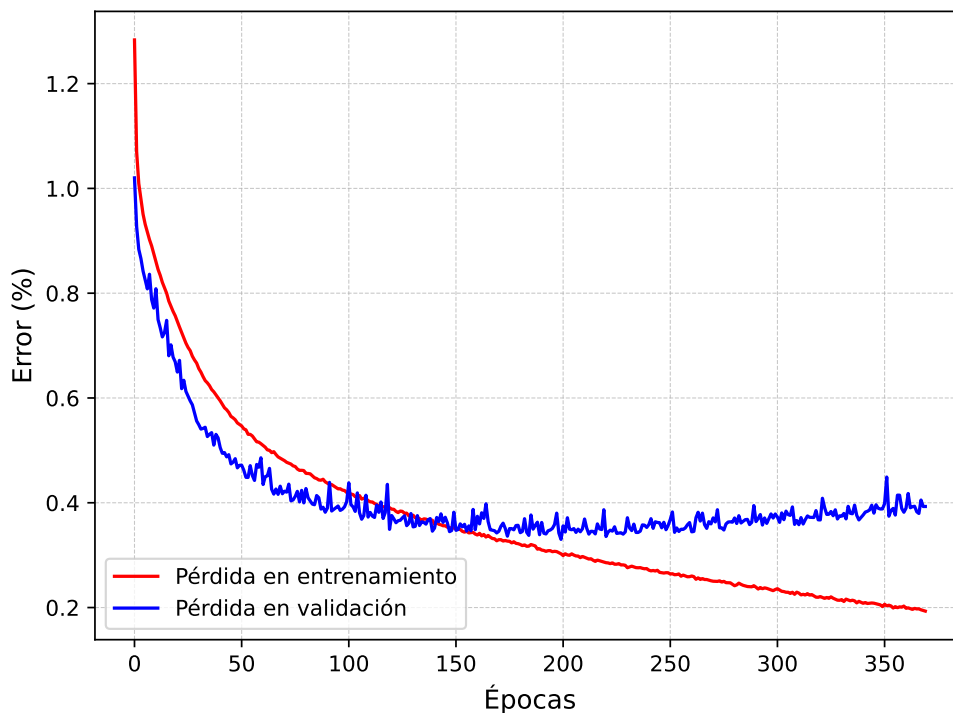


Figura 4.13: Gráfica de la pérdida en entrenamiento y validación

En conjunto, las figuras 4.12 y 4.13 evidencian un comportamiento normal de una red neuronal bien entrenada, aunque con indicios de sobreajuste en las últimas etapas del entrenamiento, detenido por las técnicas de *early stopping*.

Se obtuvo la matriz de confusión para evaluar el rendimiento del modelo multiclase, en la figura 4.14 se muestra la comparación de las predicciones del modelo con las clases reales, permitiendo de esta manera analizar en detalle los errores y aciertos de la clasificación. La clase 4, correspondiente al movimiento de extensión de los dedos de la mano fue el que más acertó al clasificarlo y el movimiento de flexión del pulgar el que obtuvo menos aciertos.

En la matriz, las filas representan las etiquetas correctas de los datos, mientras que las columnas representan las clases predichas por el modelo. Cada celda de la matriz muestra la cantidad de veces que una clase real fue clasificada como cierta clase predicha. En la diagonal principal de la matriz se muestran los casos en los que el modelo clasificó correctamente, los valores fuera de la diagonal representan los errores de clasificación.

En la primera fila, correspondiente a la clase 1, se observa que el modelo clasificó correctamente 5,737 instancias. Sin embargo, cometió errores al predecir 158 veces que se trataba de la clase 2, 10 veces la clase 3, 133 veces la clase 4 y 844 veces la clase 5. La clase 5 es la que más confusión genera para la clase 1, ya que en ambos movimientos, gesto de puño y flexor de los dedos medios se activa mayormente el músculo flexor común.

El movimiento de flexión del pulgar, fue el de menor sensibilidad, la clase 2 tuvo 5,961 aciertos en su predicción, sin embargo cometió errores, clasificándola erróneamente como clase 1 en 158 casos, como clase 3 en 436 casos, como clase 4 en 338 casos y como clase 5 en 63 casos. Aunque la mayoría de las predicciones son correctas, existe confusión con la clase 3, el estado de reposo.

Lo que sugiere que el sensor del flexor del pulgar siendo uno de los músculos más pequeños en el antebrazo, registró voltajes máximos de 1 v, haciendo que las señales obtenidas de la acción de reposo puedan confundirse.

En la clase 3, se obtuvo una cantidad de 6,462 clasificaciones correctas, con solo 5 errores en la predicción de la clase 1. Sin embargo, el modelo confundió la clase 3 con la clase 2 en 397 casos, con la clase 4 en 72 casos y con la clase 5 en 18 casos. A pesar de estos errores, esta es la clase con menos errores en general. Lo cual se puede atribuir a las señales de bajo voltaje que se obtienen al no ejercer movimiento de los músculos.

Para la el movimiento de extensión de los dedos, la clase 4, se lograron 6,520 predicciones correctas, con 95 casos clasificados erróneamente como clase 1, 236 como clase 2, 95 como clase 3 y 44 como clase 5.

Finalmente, la clase 5 tuvo 6,226 aciertos en su clasificación, presentando desaciertos con la clase 1 en 657 casos, con la clase 2 en 56 casos, con la clase 3 en 11 casos y con la clase 4 en 68 casos. El mayor problema de clasificación en esta categoría es con la clase 1, como ocurre en la clasificación del movimiento de gesto de puño, lo que sugiere que comparen patrones similares en la representación de los datos al activar el mismo músculo para la acción.

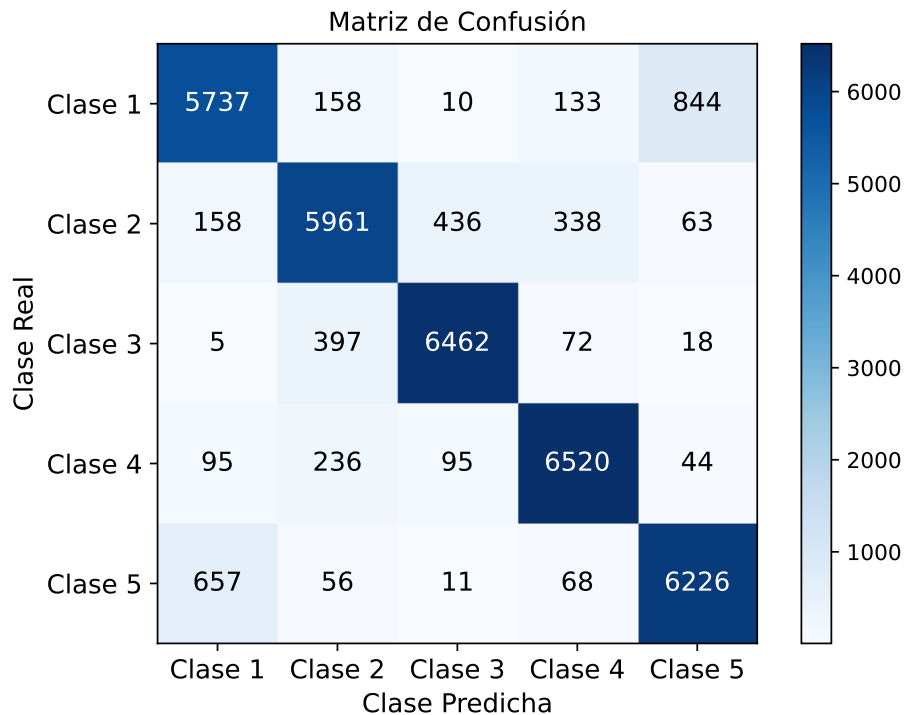


Figura 4.14: Matriz de confusión para las 5 clases.

Dado que tres de los movimientos implican la flexión de los dedos y un cuarto en el que no hay actividad muscular, el movimiento de extensión de los dedos resulta más fácil de diferenciar. Esto se debe a que el músculo extensor de los dedos se activa principalmente durante este movimiento y solo muestra una ligera participación en la flexión de los tres dedos medios. Los errores en la clasificación, como la confusión entre la clase 1 y la clase 5, y entre la clase 2 y la clase 3, se deben

a la activación de los mismos músculos en los movimientos (en el gesto de puño y flexión de los tres dedos) o el bajo voltaje producido por los músculos (en la flexión del pulgar y la acción de descanso).

Con base en la matriz de confusión se obtuvieron los casos verdaderos positivos y negativos, así como los casos falsos negativos y positivos de cada una de las 5 clases, estas variables se utilizaron para obtener las métricas de evaluación del modelo con las ecuaciones 2.16, 2.17 y 2.18.

Tabla 4.2: Métricas de evaluación del modelo por clase

Clase	Movimiento	Precisión	Sensibilidad	Especificidad
1	Gesto puño	94.08 %	83.36 %	96.72 %
2	Flexión pulgar	94.71 %	85.70 %	96.96 %
3	Descanso	97.00 %	92.92 %	98.02 %
4	Extensión dedos	96.89 %	93.28 %	97.80 %
5	Flexión 3 dedos medios	94.94 %	88.71 %	96.51 %
	Promedio	95.52 %	88.79 %	97.20 %

Obteniendo como resultado una eficiencia del 93.84 % cumpliendo con la hipótesis planteada.

4.5. Interfaz gráfica

Se diseñó una interfaz gráfica que permite visualizar y analizar señales EMG en función del movimiento realizado. El usuario ingresa un número entre 0 y 34799, correspondiente a una señal específica, y el sistema predice el movimiento asociado utilizando el modelo obtenido. La interfaz genera tres gráficas, cada una representando los datos de un sensor EMG distinto, comparando la señal real (en azul) con la predicción del modelo (en rojo), como se observa en 4.15.

Además, muestra en pantalla el resultado de la clasificación, indicando si la predicción fue correcta o incorrecta junto con los nombres del movimiento real y el predicho. Esto proporciona una manera rápida e intuitiva de evaluar el desempeño del modelo y la correspondencia entre las señales EMG y los movimientos clasificados.

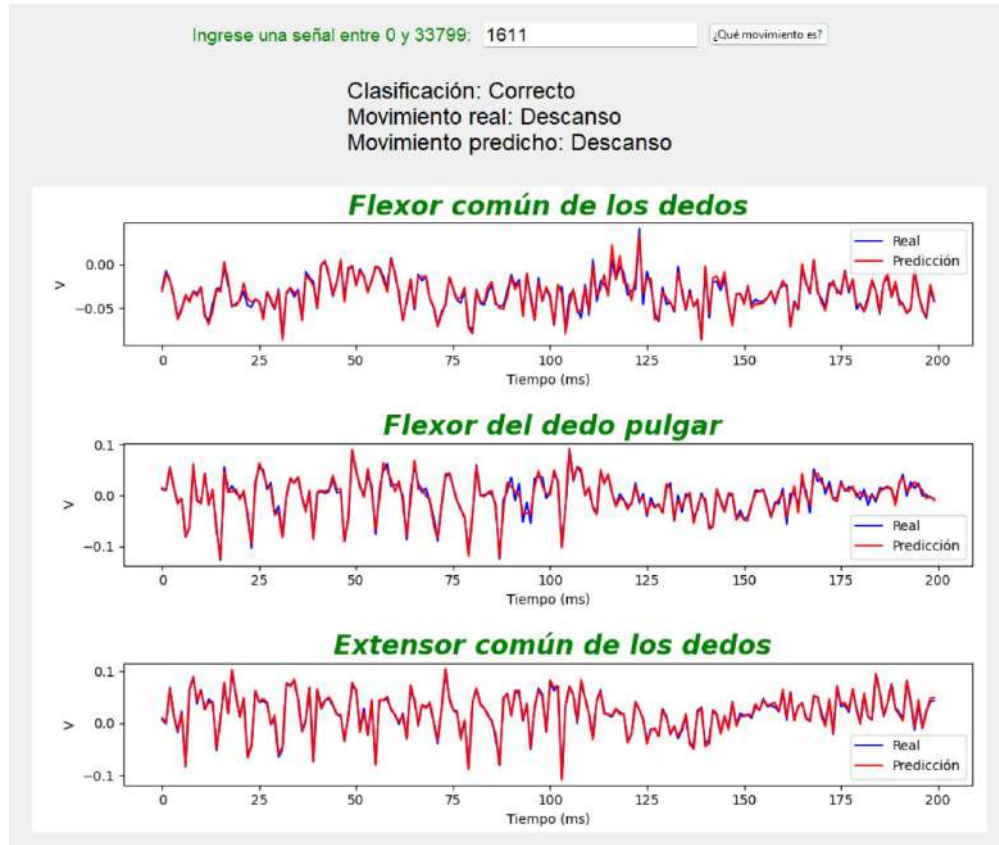


Figura 4.15: Interfaz gráfica.

4.6. Discusión

En este trabajo se exploró el uso de un autoencoder para la replicación de señales EMG y la extracción de características latentes con el fin de mejorar la clasificación de movimientos de la mano. A diferencia de otros estudios que emplean directamente las señales crudas o solo con un procesamiento digital previo, esta metodología permite obtener una representación compacta y no supervisada de la información relevante en las señales EMG.

Los resultados mostraron que los autoencoders fueron capaces de replicar las señales con un error mínimo, lo que sugiere que la estructura latente capturada contiene la información esencial de la actividad muscular aprovechada por el clasificador. Además, la representación comprimida reduce la dimensionalidad de los datos, lo que potencialmente disminuye el tiempo de entrenamiento y mejora la generalización del modelo.

Esto es consistente con trabajos previos que han demostrado la efectividad de autoencoders en la reducción de ruido y la extracción de características relevantes en señales.

Los estudios previos, demostraron que el uso de RNN, particularmente arquitecturas basadas en LSTM y GRU, han sido altamente efectivas en la clasificación de señales EMG para la detección de gestos de la mano. Por esta razón, se optó por un autoencoder con arquitectura LSTM y GRU, ya que están diseñadas para manejar dependencias temporales, lo que las hace útiles para datos que tienen una secuencia de estados que dependen unos de otros a lo largo del tiempo.

En [11], se evaluaron 16 modelos diferentes, concluyendo que una red con unidades LSTM bidireccionales obtuvo la mejor precisión en la clasificación multiclase de gestos de la mano. Sin embargo, al probar la arquitectura bidireccional los resultados no mejoraron, esto puede ser debido a que las señales EMG tienen una dirección temporal clara, por lo que una red unidireccional ya captura la información relevante, además de que el desgaste computacional se veía comprometido.

El vector latente obtenido del autoencoder tiene una representación comprimida y fija de la señal, un vector de características para clasificarla directamente. Por ello, las capas densas son adecuadas, ya que se enfocan en mapear un vector de entrada a una clase específica de forma eficiente. En la clasificación, no se busca capturar la secuencia temporal de la señal de la misma manera, sino, las características que se pueden utilizar para identificar la clase de la señal.

Los resultados muestran una precisión comparable a los estudios revisados, evidenciando la efectividad de las capas para la clasificación de señales EMG. Sin embargo, al igual que lo sugerido por [16], se identificó que el incremento en la cantidad de canales y la optimización de los hiperparámetros pueden mejorar el rendimiento del modelo. Además, es posible que la incorporación de un mecanismo de atención, como en [11], pueda contribuir a una mejor representación de las características temporales de la señal EMG.

En la tabla 4.3 se observa la comparación de los trabajos anteriores con el presente, destacando la cantidad de sujetos, movimientos y la precisión máxima.

En [13], se alcanzó una precisión del 95 % utilizando la base de datos DualMyo, que incluye únicamente un sujeto y ocho gestos. De manera similar, [5] reportó una precisión del 99.6 % utilizando redes LSTM con datos también de un solo participante. Estos estudios, aunque útiles como pruebas de concepto, no enfrentan la variabilidad entre diferentes sujetos, que es un desafío real en aplicaciones clínicas o comerciales.

En contraste, el presente trabajo utiliza una base de datos recolectada con 20 personas diferentes, lo cual introduce una mayor variabilidad fisiológica y, por tanto, representa un escenario más realista para aplicaciones prácticas. A pesar de esta complejidad, se logró una precisión promedio competitiva, lo cual resalta la robustez del modelo propuesto.

Además, otros estudios como [12] y [14] utilizaron bases de datos con 6 y 18 sujetos respectivamente, pero con configuraciones de sensores más complejas, como EMG de alta densidad o configuraciones no portátiles. En cambio, en este trabajo se empleó una configuración simple de tres sensores, más adecuada para futuras aplicaciones en tiempo real o en sistemas embebidos.

Tabla 4.3: Comparación de los trabajos previos con el actual.

Autor	Sujetos	Movimientos realizados	Precisión máxima
[11]	40	50	89.5 %
[12]	6	7	95.52 %
[13]	1	58	95 %
[5]	1	8	99.6 %
[15]	1	5	90.7 %
[16]	6	5	97.25 %
[18]	9	4	98.46 %
Este trabajo	20	3	97 %

En comparación con otros trabajos que utilizan redes neuronales profundas para la clasificación directa de EMG, esta metodología basada en autoencoders y clasificador con capas densas, presenta

la ventaja de reducir la cantidad de datos requeridos para el entrenamiento del clasificador, al tiempo que mantiene un rendimiento competitivo.

Conclusiones

En este estudio, se ha demostrado que el uso de autoencoders, específicamente con capas LSTM y GRU, es eficaz para la compresión y extracción de características de señales EMG en la clasificación de movimientos de la mano. La segmentación en ventanas superpuestas de corta duración se ha mostrado como una estrategia adecuada para capturar patrones temporales relevantes de la señal EMG, mejorando la precisión en la clasificación de los movimientos musculares.

Los resultados también confirman que la combinación de técnicas avanzadas como las redes neuronales recurrentes en el autoencoder y la segmentación en ventanas contribuye significativamente a la precisión y eficiencia en el procesamiento de señales musculares.

Sin embargo, es fundamental destacar que la organización adecuada de la base de datos juega un papel crucial en el desempeño del modelo. La aleatorización de los datos es esencial, ya que una mala organización puede comprometer los resultados, introduciendo sesgos en la validación y entrenamiento, lo que afectaría negativamente la capacidad del modelo para generalizar en escenarios reales.

Además, una correcta toma de señales es crucial para asegurar la calidad de los datos; un protocolo preciso y estandarizado con todos los participantes es necesario para minimizar el margen de error y asegurar la solidez del modelo. Asegurar que este protocolo sea replicado consistentemente en todos los participantes garantizará que los resultados sean más fiables y generalizables.

5.1. Trabajos Futuros

El presente trabajo propone el desarrollo de un sistema de control de dirección para una silla de ruedas basado en señales EMG. Hasta el momento, se cuenta con un dispositivo de adquisición de señales EMG, y modelos de predicción y clasificación alcanzando una eficiencia del 93 %. Como siguiente etapa, se propone la instrumentación de la silla de ruedas y la integración del sistema de control. Permitiendo que la silla ejecute cinco acciones: avanzar, retroceder, girar a la izquierda, girar a la derecha y permanecer estática, cada una de ellas asociada a un movimiento específico de la mano.

En la metodología, se experimentó con diferentes porcentajes de traslape (70 %, 75 % y 80 %), observándose un impacto significativo en los resultados. Con base en estos hallazgos, se considera que el intervalo de 200 ms podría extenderse hasta 300 ms en estudios futuros.

De igual manera, se podría evaluar la integración de las características latentes obtenidas de los autoencoders con métodos tradicionales de clasificación, como las máquinas de soporte vectorial,

para explorar si esta combinación mejora el rendimiento de los sistemas en tiempo real.

Posteriores estudios podrían explorar el uso de autoencoders variantes, como los autoencoders variacionales y los autoencoders convolucionales, con el fin de mejorar aún más la extracción de características relevantes de las señales EMG.

La combinación de estos autoencoders con métodos clásicos de extracción de características, como análisis de frecuencias o transformadas, podría mejorar la capacidad de clasificación, especialmente en contextos complejos y ruidosos.

Bibliografía

- [1] INEGI.org.mx, “Población. discapacidad,” 2020.
- [2] H. Ghapanchizadeh, S. A. Ahmad, A. J. Ishak, and M. S. Al-Quraishi, “Review of surface electrode placement for recording electromyography signals.,” *Biomedical Research (0970-938X)*, vol. 28, 2017.
- [3] F. B. Stulen and C. J. De Luca, “Frequency parameters of the myoelectric signal as a measure of muscle conduction velocity,” *IEEE Transactions on Biomedical Engineering*, no. 7, pp. 515–523, 1981.
- [4] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” *nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [5] T. M. Bittibssi, M. A. Genedy, S. A. Maged, *et al.*, “semg pattern recognition based on recurrent neural network,” *Biomedical Signal Processing and Control*, vol. 70, p. 103048, 2021.
- [6] M. C. Cabedio and M. Carnaghi, “Espectrogramas de registros de ballenas barbadas sintetizados a partir de arquitecturas de autoencoders: Cae, vae y cae-lstm,” *Elektron: ciencia y tecnología en la electrónica de hoy*, vol. 6, no. 2, pp. 129–134, 2022.
- [7] O. C. Carrasco, “Modelos generativos en el aprendizaje automático y su aplicación a la generación de imágenes digitales,” *Journal Boliviano de Ciencias*, vol. 17, no. 51, pp. 79–109, 2021.
- [8] S. A. N. Caro, *USO DE AUTOENCODER PARA LA SEGMENTACIÓN Y SIMULACIÓN DE FIBRAS CEREBRALES*. PhD thesis, Universidad de Concepción, 2023.
- [9] P. B. Weerakody, K. W. Wong, G. Wang, and W. Ela, “A review of irregular time series data handling with gated recurrent neural networks,” *Neurocomputing*, vol. 441, pp. 161–178, 2021.
- [10] N. Jaouedi, N. Boujnah, and M. S. Bouhleb, “Deep learning approach for human action recognition using gated recurrent unit neural networks and motion analysis,” *Journal of Computer Science*, vol. 15, no. 7, pp. 1040–1049, 2019.
- [11] A. Samadani, “Gated recurrent neural networks for emg-based hand gesture classification. a comparative study,” in *2018 40th annual international conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 1–4, IEEE, 2018.

- [12] K. Vorsevych, M. F. Goethel, J. Mrozowski, J. Awrejcewicz, and M. Bezuglyi, “Fingers movements control system based on artificial neural network model,” *Radioelectronics and Communications Systems*, vol. 62, no. 1, pp. 23–33, 2019.
- [13] M. Simão, P. Neto, and O. Gibaru, “Emg-based online classification of gestures with recurrent neural networks,” *Pattern Recognition Letters*, vol. 128, pp. 45–51, 2019.
- [14] D. Xiong, D. Zhang, X. Zhao, and Y. Zhao, “Hand gesture recognition using instant high-density emg graph via deep learning method,” in *2020 Chinese Automation Congress (CAC)*, pp. 5143–5148, IEEE, 2020.
- [15] A. Toro-Ossaba, J. C. Tejada, S. Rúa, and A. López-González, “A proposal of bioinspired soft active hand prosthesis,” *Biomimetics*, vol. 8, no. 1, p. 29, 2023.
- [16] M. T. N. Truong, A. E. A. Ali, D. Owaki, and M. Hayashibe, “Emg-based estimation of lower limb joint angles and moments using long short-term memory network,” *Sensors*, vol. 23, no. 6, 2023.
- [17] H. A. Hosni Mahmoud and E. Alabdulkreem, “Bidirectional neural network model for glaucoma progression prediction,” *Journal of Personalized Medicine*, vol. 13, no. 3, 2023.
- [18] M. Aviles, J. M. Alvarez-Alvarado, J.-B. Robles-Ocampo, P. Y. Sevilla-Camacho, and J. Rodríguez-Reséndiz, “Optimizing rnns for emg signal classification: A novel strategy using grey wolf optimization,” *Bioengineering*, vol. 11, no. 1, p. 77, 2024.
- [19] Organización Mundial de la Salud, “Tecnología de asistencia,” 2023. Accesado el 21 de febrero de 2024.
- [20] I. N. de Desarrollo Social, “Autonomía a personas que han sufrido amputaciones,” 2017.
- [21] Informes de Expertos, “Mercado latinoamericano de silla de ruedas asistida por energía: Análisis y pronóstico (2023-2028),” 2023.
- [22] Mordor Intelligence, “Mercado de sillas de ruedas activas - Tamaño, participación, crecimiento, tendencias y análisis de pronósticos (2023-2028),” 2023.
- [23] OMS., *Trastornos neurológicos: desafíos para la salud pública*. Organización Mundial de la Salud, 2006.
- [24] S. E. H. Mendoza *et al.*, “Instrumentación y control para una silla de ruedas eléctrica con comunicación verbal,” 2023.
- [25] G. M. N. Silva and T. L. C. Miranda, “Redes neuronales y biopotenciales musculares: Hacia un control más natural de prótesis mioeléctricas,” *INGENIERÍA INVESTIGA*, vol. 6, 2024.
- [26] C. A. S. Rico, O. J. S. Sierra, and J. A. M. Hermosillo, “Diseño ergonómico de una prótesis biónica para miembro superior controlada por señales electromiográficas,” *REVISTA COLOMBIANA DE TECNOLOGIAS DE AVANZADA (RCTA)*, vol. 1, no. 43, pp. 99–109, 2024.
- [27] J. E. Hall, *Guyton y Hall. Tratado de fisiología médica*. Elsevier Health Sciences, 2011.

- [28] R. L. Lieber, *Skeletal muscle structure, function, and plasticity*. Lippincott Williams & Wilkins, 2002.
- [29] D. Lovely, “The origins and nature of the myoelectric signal,” in *Powered Upper Limb Prostheses: Control, Implementation and Clinical Application*, pp. 17–33, Springer, 2004.
- [30] *Una introducción práctica a la electromiografía kinesiología*, 2005.
- [31] E. Altamira, M. Mayorga, and G. Paredes, “Electromiograma emg,” 2017.
- [32] D. D. Proaño Guevara, “Sistema de adquisición de señales emg de superficie multicanal para prótesis de miembro superior,” B.S. thesis, 2019.
- [33] B. Karlik, “Machine learning algorithms for characterization of emg signals,” *International Journal of Information and Electronics Engineering*, vol. 4, no. 3, p. 189, 2014.
- [34] A. B. Landete, “Mitofagia en enfermedades neurodegenerativas,”
- [35] H. J. Hermens, B. Freriks, C. Disselhorst-Klug, and G. Rau, “Development of recommendations for semg sensors and sensor placement procedures,” *Journal of electromyography and Kinesiology*, vol. 10, no. 5, pp. 361–374, 2000.
- [36] R. Merletti, A. Botter, A. Troiano, E. Merlo, and M. A. Minetto, “Technology and instrumentation for detection and conditioning of the surface electromyographic signal: state of the art,” *Clinical biomechanics*, vol. 24, no. 2, pp. 122–134, 2009.
- [37] Y. Fukuoka, K. Miyazawa, H. Mori, M. Miyagi, M. Nishida, Y. Horiuchi, A. Ichikawa, H. Hoshino, M. Noshiro, and A. Ueno, “Development of a compact wireless laplacian electrode module for electromyograms and its human interface applications,” *Sensors*, vol. 13, no. 2, pp. 2368–2383, 2013.
- [38] A. Ueno, Y. Uchikawa, and M. Noshiro, “A capacitive sensor system for measuring laplacian electromyogram through cloth: a pilot study,” in *2007 29th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 5731–5734, IEEE, 2007.
- [39] M. Hakonen, H. Piitulainen, and A. Visala, “Current state of digital signal processing in myoelectric interfaces and related applications,” *Biomedical Signal Processing and Control*, vol. 18, pp. 334–359, 2015.
- [40] M. R. Avilés *et al.*, “Clasificación de señales mioeléctricas por medio de algoritmos genéticos y máquinas de soporte de vectores,” 2022.
- [41] I. S. Dhindsa, R. Agarwal, and H. S. Ryaith, “Performance evaluation of various classifiers for predicting knee angle from electromyography signals,” *Expert Systems*, vol. 36, no. 3, p. e12381, 2019. e12381 EXSY-Jan-18-002.R4.
- [42] A. Vijayvargiya, V. Gupta, R. Kumar, N. Dey, and J. M. R. Tavares, “A hybrid wd-eemd semg feature extraction technique for lower limb activity recognition,” *IEEE Sensors Journal*, vol. 21, no. 18, pp. 20431–20439, 2021.
- [43] M. A. Oskoei and H. Hu, “Myoelectric control systems—a survey,” *Biomedical signal processing and control*, vol. 2, no. 4, pp. 275–294, 2007.

- [44] D. C. Toledo-Pérez, J. Rodríguez-Reséndiz, R. A. Gómez-Loenzo, and J. Jauregui-Correa, "Support vector machine-based emg signal classification techniques: A review," *Applied Sciences*, vol. 9, no. 20, p. 4402, 2019.
- [45] C. Fang, B. He, Y. Wang, J. Cao, and S. Gao, "Emg-centered multisensory based technologies for pattern recognition in rehabilitation: state of the art and challenges," *Biosensors*, vol. 10, no. 8, p. 85, 2020.
- [46] G. Li, Y. Li, L. Yu, and Y. Geng, "Conditioning and sampling issues of emg signals in motion recognition of multifunctional myoelectric prostheses," *Annals of biomedical engineering*, vol. 39, pp. 1779–1787, 2011.
- [47] W. H. Hayt, J. E. Kemmerly, S. M. Durbin, and J. D. Philips, *Análisis de circuitos en ingeniería*. McGraw Hill, 2012.
- [48] F. Miyara, "Filtros activos," *Cátedra de Electrónica III FCEIA-UNR. Rosario*, 2004.
- [49] E. C. Ifeachor and B. W. Jervis, *Digital signal processing: a practical approach*. Pearson Education, 2002.
- [50] M. Boyer, L. Bouyer, J.-S. Roy, and A. Campeau-Lecours, "Reducing noise, artifacts and interference in single-channel emg signals: A review," *Sensors*, vol. 23, no. 6, p. 2927, 2023.
- [51] G. Li, Y. Li, Z. Zhang, Y. Geng, and R. Zhou, "Selection of sampling rate for emg pattern recognition based prosthesis control," in *2010 Annual International Conference of the IEEE Engineering in Medicine and Biology*, pp. 5058–5061, IEEE, 2010.
- [52] Y. M. Reyes, "Adquisición de características de señales mioeléctricas para uso en prótesis," 2012.
- [53] T. R. Farrell, "Determining delay created by multifunctional prosthesis controllers," *Journal of rehabilitation research and development*, vol. 48, no. 6, p. xxi, 2011.
- [54] K. Englehart and B. Hudgins, "A robust, real-time control scheme for multifunction myoelectric control," *IEEE transactions on biomedical engineering*, vol. 50, no. 7, pp. 848–854, 2003.
- [55] M. A. Oskoei and H. Hu, "Support vector machine-based classification scheme for myoelectric control applied to upper limb," *IEEE transactions on biomedical engineering*, vol. 55, no. 8, pp. 1956–1965, 2008.
- [56] S. J. Russell and P. Norvig, *Artificial intelligence a modern approach*. London, 2010.
- [57] S. Shalev-Shwartz and S. Ben-David, *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.
- [58] M. T. Hagan, H. B. Demuth, M. H. Beale, and O. De Jesús, "Neural network design. 2nd edition," *Oklahoma: Martin Hagan*, 2014.
- [59] R. F. López and J. M. F. Fernández, *Las redes neuronales artificiales*. Netbiblo, 2008.
- [60] X. B. Olabe, "Redes neuronales artificiales y sus aplicaciones," *Publicaciones de la Escuela de Ingenieros*, 1998.

- [61] C. H. M. Chiok, “Modelos de regresión lineal con redes neuronales,” in *Anales científicos*, vol. 75, pp. 253–260, Universidad Nacional Agraria La Molina, 2014.
- [62] H. Allende, C. Moraga, and R. Salas, “Artificial neural networks in time series forecasting: A comparative analysis,” *Kybernetika*, vol. 38, no. 6, pp. 685–707, 2002.
- [63] X. Basogain, “Redes neuronales artificiales y sus aplicaciones,” *Dpto. Ingeniería de Sistemas y Automática, Escuela Superior de Ingeniería Bilbao. Open Course Ware.*[En línea] disponible en http://ocw.ehu.es/enseñanzas-tecnicas/redes-neuronales-artificiales-y-sus-aplicaciones/Course_listing. [Consultada 20-09-2012], 2008.
- [64] A. Severyn and A. Moschitti, “Learning to rank short text pairs with convolutional deep neural networks,” in *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, pp. 373–382, 2015.
- [65] Á. Artola Moreno, “Clasificación de imágenes usando redes neuronales convolucionales en python,” 2019.
- [66] V. S. Tomar, *A critical evaluation of activation functions for autoencoder neural networks*. PhD thesis, Dublin, National College of Ireland, 2022.
- [67] M. Lee, “Gelu activation function in deep learning: a comprehensive mathematical analysis and performance,” *arXiv preprint arXiv:2305.12073*, 2023.
- [68] A. Nguyen, K. Pham, D. Ngo, T. Ngo, and L. Pham, “An analysis of state-of-the-art activation functions for supervised deep neural network,” in *2021 International conference on system science and engineering (ICSSE)*, pp. 215–220, IEEE, 2021.
- [69] F. Chollet *et al.*, “Keras: The python deep learning framework,” 2015.
- [70] Y. A. Buitrago López *et al.*, “Pronóstico del covid-19 en colombia utilizando redes neuronales recurrentes con celdas de gran memoria de corto plazo y unidades recurrentes cerradas,” B.S. thesis, 2022.
- [71] A. E. Repetur, “Redes neuronales artificiales,” *Buenos Aires: Universidad Nacional del Centro de la Provincia de Buenos Aires*, 2019.
- [72] J. M. Alvarado Carrera, “Clasificación de sílabos académicos en base a redes neuronales de aprendizaje profundo,” B.S. thesis, Universidad del Azuay, 2018.
- [73] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.
- [74] Y. Bengio, P. Simard, and P. Frasconi, “Learning long-term dependencies with gradient descent is difficult,” *IEEE transactions on neural networks*, vol. 5, no. 2, pp. 157–166, 1994.
- [75] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [76] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, “On the properties of neural machine translation: Encoder-decoder approaches,” *arXiv preprint arXiv:1409.1259*, 2014.

- [77] Y. Yu, X. Si, C. Hu, and J. Zhang, “A review of recurrent neural networks: Lstm cells and network architectures,” *Neural computation*, vol. 31, no. 7, pp. 1235–1270, 2019.
- [78] R. S. Guevara Cruz and C. A. Delrieux, “Aplicación de redes neuronales densas y convolucionales para detección de covid_19 en imágenes de rayos x,” 2023.
- [79] L. Rascón-Madrigal, M. Sinecio-Sidrian, J. Mejía-Muñoz, J. Díaz-Román, I. Canales-Valdiviezo, and A. Botello-Arredondo, “Estimation of hand-grip intention: Cylindrical, spherical and hook using deep neural networks,” *Revista mexicana de ingeniería biomédica*, vol. 41, no. 1, pp. 117–127, 2020.
- [80] K. Berahmand, F. Daneshfar, E. S. Salehi, Y. Li, and Y. Xu, “Autoencoders and their applications in machine learning: a survey,” *Artificial Intelligence Review*, vol. 57, no. 2, p. 28, 2024.
- [81] P. Li, Y. Pei, and J. Li, “A comprehensive survey on design and application of autoencoder in deep learning,” *Applied Soft Computing*, vol. 138, p. 110176, 2023.
- [82] G. F. González Estrada, “Robust-neurobiometrics: sistema robusto de identificación biométrica basado en señales de electroencefalograma (eeg),” 2024.
- [83] A. H. Mirza and S. Cosan, “Computer network intrusion detection using sequential lstm neural networks autoencoders,” in *2018 26th Signal Processing and Communications Applications Conference (SIU)*, pp. 1–4, 2018.
- [84] P. Mobtahej, X. Zhang, M. Hamidi, and J. Zhang, “An lstm-autoencoder architecture for anomaly detection applied on compressors audio data,” *Computational and Mathematical Methods*, vol. 2022, no. 1, p. 3622426, 2022.
- [85] I. Rodríguez Ferrero, *Uso de autoencoders en la compresión de imágenes*. PhD thesis, Universitat Politècnica de València, 2022.
- [86] O. D. Castaño Palacio, “Identificación de tareas motoras a través de algoritmos profundos de clasificación mediante la aplicación de autoencoders variacionales,” 2023.
- [87] D. Chicco, M. J. Warrens, and G. Jurman, “The coefficient of determination r-squared is more informative than smape, mae, mape, mse and rmse in regression analysis evaluation,” *Peerj computer science*, vol. 7, p. e623, 2021.
- [88] C. Chen, J. Twycross, and J. M. Garibaldi, “A new accuracy measure based on bounded relative error for time series forecasting,” *PloS one*, vol. 12, no. 3, p. e0174202, 2017.
- [89] D. M. Hawkins, S. C. Basak, and D. Mills, “Assessing model fit by cross-validation,” *Journal of chemical information and computer sciences*, vol. 43, no. 2, pp. 579–586, 2003.
- [90] S. Arlot and A. Celisse, “A survey of cross-validation procedures for model selection,” 2010.
- [91] L. Pérez-Planells, J. Delegido, J. P. Rivera-Caicedo, and J. Verrelst, “Análisis de métodos de validación cruzada para la obtención robusta de parámetros biofísicos,” *Revista de teledetección*, no. 44, pp. 55–65, 2015.

- [92] A. J. Llantoy Sánchez, “Diseño e implementación del sistema electrónico para una prótesis transradial mioeléctrica,”
- [93] D. ir. H.J. Hermens and ir. B Freriks, “Semg sensors,” 1996.
- [94] C. M. Durán Acevedo and A. L. Jaimes Mogollón, “Optimización y clasificación de señales emg a través de métodos de reconocimiento de patrones,” *Iteckne*, vol. 10, no. 1, pp. 67–76, 2013.
- [95] G. Naik, *Biomedical Signal Processing*. Springer, 2020.
- [96] S. learn developers, “sklearn.model_selection.train_test_split,” 2024. Último acceso: 30 de marzo de 2025.