

Diseño e Implementación de una Prótesis Biónica Controlada por Señales EMG mediante Machine Learning

Isaí Pérez Herrera
*Estudiante de Ingeniería Mecatrónica
Instituto Tecnológico y de Estudios
Superiores de Monterrey
Tampico, Tamaulipas, México
A01412968@tec.mx*

Diego Antonio Figueroa Bernal
*Estudiante de Ingeniería Mecatrónica
Instituto Tecnológico y de Estudios
Superiores de Monterrey
Tampico, Tamaulipas, México
A01412125@tec.mx*

José Ángel Briones Cárdenas
*Estudiante de Ingeniería Mecatrónica
Instituto Tecnológico y de Estudios
Superiores de Monterrey
Tampico, Tamaulipas, México
A01412078@tec.mx*

Abstract— Este proyecto presenta una prótesis de mano biónica controlada por señales electromiográficas (EMG), capaz de reconocer tres estados musculares de la mano: abierto, cerrado y en reposo. El objetivo es diseñar una prótesis funcional, con un sistema de control robusto sin calibración personalizada. Se recolectaron y etiquetaron datos EMG para entrenar distintos modelos de clasificación, seleccionando Random Forest por ser el que alcanzó mayor precisión (83 %). Las características extraídas incluyeron el valor absoluto medio, desviación estándar y valor máximo. El sistema se divide en dos subsistemas conectados por TCP/IP, dado que el sensor MyoWare 2.0 es sensible al ruido eléctrico y no puede operar cerca de motores ni de computadoras conectadas a corriente. Un Arduino transmite las señales EMG a una PC cliente que clasifica el gesto muscular y envía la predicción a un servidor. Allí, otro Arduino, conectado a un controlador PCA9685, acciona los servomotores de la prótesis. El sistema fue validado en tiempo real y demostró un tiempo de respuesta de aproximadamente un segundo, lo cual confirma su viabilidad como solución de bajo costo y alta adaptabilidad para el control mioeléctrico.

Keywords— *Prótesis biónica, señales electromiográficas, Machine Learning, control mioeléctrico, Random Forest, Arduino, PCA9685.*

I. INTRODUCCIÓN

En los últimos años, el desarrollo de prótesis de extremidades superiores ha avanzado significativamente gracias a la integración de sensores biomédicos, sistemas embebidos y algoritmos de inteligencia artificial. Sin embargo, muchas de estas soluciones aún presentan limitaciones importantes en términos de costo, portabilidad y necesidad de calibración personalizada. Esto dificulta su adopción en contextos de bajo presupuesto o por parte de usuarios que requieren una solución accesible y funcional para el uso cotidiano.

Las señales electromiográficas (EMG), generadas por la actividad eléctrica de los músculos, ofrecen una vía efectiva para controlar dispositivos protésicos de manera intuitiva. No obstante, estas señales presentan un alto nivel de variabilidad y sensibilidad al ruido, lo cual representa un reto técnico en entornos no controlados. El presente proyecto aborda este desafío mediante el diseño e implementación de una mano biónica controlada por EMG, utilizando algoritmos de Machine Learning para la clasificación de gestos musculares simples.

El sistema propuesto tiene como objetivo reconocer tres estados musculares de la mano —abierto, cerrado y en reposo—, sin requerir calibración personalizada. Para lograrlo, se diseñó una arquitectura dividida en dos subsistemas: uno para la adquisición de señales EMG y otro

para el control de actuadores. Esta separación responde a limitaciones físicas del sensor MyoWare 2.0, que es altamente sensible al ruido eléctrico y no puede operar en entornos compartidos con motores o equipos conectados a corriente.

Durante el desarrollo, se llevó a cabo la adquisición y etiquetado de datos EMG de usuarios sanos, seguida por la extracción de características y el entrenamiento de distintos modelos de clasificación. Tras una evaluación comparativa, se seleccionó un modelo Random Forest por su precisión superior. Las predicciones se comunican mediante el protocolo TCP/IP, lo que permite modularidad, aislamiento eléctrico y flexibilidad de implementación. Los resultados obtenidos demuestran que el sistema alcanza una precisión del 83 % y un tiempo de respuesta cercano a un segundo, posicionándolo como una alternativa viable y de bajo costo para el control de prótesis mioeléctricas.

II. MARCO TEÓRICO

A. Fundamentos de las señales EMG

La electromiografía (EMG) se refiere al registro de las señales eléctricas generadas por los músculos esqueléticos. En el caso de la EMG superficial (sEMG), los electrodos colocados sobre la piel captan una señal compuesta por la superposición de los potenciales de acción de numerosas fibras musculares subyacentes, las cuales se activan en intervalos aleatorios. La amplitud típica de estas señales varía aproximadamente entre 0 y 10 mV (valor pico a pico) antes de cualquier amplificación, concentrándose la mayor parte de su contenido de energía en frecuencias por debajo de ~500 Hz. Debido a que las señales EMG suelen estar contaminadas por ruido (eléctrico, ambiental, artefactos de movimiento, etc.), es crucial optimizar la relación señal-ruido mediante un adecuado procesamiento filtrado de la señal bruta.

B. Procesamiento de señales EMG

Filtrado: Como primer paso, la señal EMG se suele filtrar con un pasa-bandas (ej. rango 20–450 Hz) para suprimir componentes de muy baja y alta frecuencia —eliminando tanto artefactos de movimiento como ruido de alta frecuencia—, aplicándose además un filtro *notch* en 50/60 Hz para remover la interferencia de la red eléctrica. Este preprocesamiento mejora la fidelidad de la señal al reducir significativamente el ruido y las distorsiones indeseadas. En este proyecto, se utilizó el sensor MyoWare 2.0, el cual incorpora internamente un filtro pasa-bandas activo que atenúa señales fuera del rango útil, reduciendo significativamente el ruido de baja y alta frecuencia.

Extracción de características: Tras el filtrado, se calculan diversas características (features) cuantitativas de la señal con el fin de resumir su información relevante para clasificación. Comúnmente se extraen parámetros en el dominio temporal como el valor absoluto medio (MAV), la raíz cuadrática media (RMS), el número de cruces por cero (ZC), los cambios de pendiente (SSC) y la longitud de la onda (WL), entre otros. Adicionalmente, pueden obtenerse características en el dominio frecuencial (por ejemplo, frecuencia media, mediana o energía en ciertas bandas del espectro) para captar información complementaria de la señal EMG, las cuales junto a las características temporales sirven de entrada a los modelos de clasificación.

C. Modelos de clasificación utilizados

Árbol de decisión (Decision Tree): Algoritmo de aprendizaje supervisado que organiza las decisiones en una estructura ramificada. A partir de un nodo raíz inicial, los datos se dividen sucesivamente según distintos atributos o reglas, formando un árbol donde las hojas finales representan las clases predichas. Los árboles de decisión son intuitivos y fáciles de interpretar, por lo que se han utilizado ampliamente en tareas de clasificación de señales biomédicas.

Bosque aleatorio (Random Forest): Modelo de *ensemble* que combina múltiples árboles de decisión. Cada árbol del bosque se entrena con diferentes subconjuntos de datos y contribuye con una “votación” a la predicción final; de esta manera, el Random Forest logra aumentar la precisión y robustez respecto a un solo árbol individual. Esta técnica tiende a reducir el sobreajuste y ha demostrado buen rendimiento en la clasificación de señales EMG en diversas aplicaciones.

Máquina de vectores de soporte (SVM): Clasificador supervisado que busca un hiperplano óptimo que separe los datos de distintas clases con el mayor margen posible. Las SVM han sido muy empleadas en el reconocimiento de patrones EMG y a menudo obtienen una elevada exactitud, superando a otros algoritmos, gracias a su alto poder de generalización en espacios de alta dimensión.

III. METODOLOGÍA

A. Diseño del sistema

El sistema desarrollado se compone de dos subsistemas principales: uno de lectura y otro de control, comunicados mediante el protocolo TCP/IP en red local. Esta arquitectura responde a limitaciones eléctricas del sensor EMG MyoWare 2.0, que lo hacen sensible al ruido generado por motores y equipos conectados a tierra.

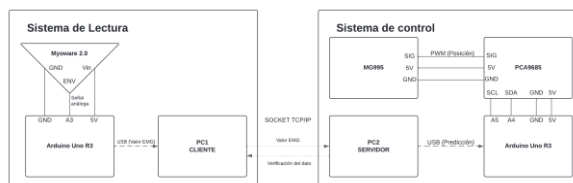


Fig. 1. Sistema de control y lectura

El subsistema de lectura incluye el sensor EMG conectado a un Arduino Uno R3, el cual transmite los datos crudos a una

PC mediante comunicación serial. En la computadora cliente, se procesan las señales, se obtienen sus características y se ejecuta el modelo de clasificación en tiempo real. La predicción resultante se envía vía socket a una segunda PC, que actúa como servidor.

El subsistema de control está compuesto por una segunda PC conectada a otro Arduino Uno R3. Este Arduino está enlazado al controlador PCA9685, el cual gestiona los servomotores de la prótesis. Una vez obtenida la predicción en la computadora cliente, esta se transmite al servidor mediante socket TCP. El servidor reenvía el valor al Arduino Uno R3 conectado por USB, el cual interpreta la clase recibida (0: reposo, 1: cerrado, 2: abierto) y calcula el pulso PWM correspondiente para cada servomotor.

Para controlar los motores, se utilizó el controlador PCA9685, que permite generar señales PWM precisas a través de I2C. Cada servomotor fue configurado con posiciones específicas para los tres gestos definidos. Esta arquitectura modular permite un tiempo de respuesta aproximado de un segundo, desde la activación muscular hasta el movimiento completo de la prótesis.

Tabla I. COMPONENTES ELÉCTRICOS Y ELECTRÓNICOS

COMPONENTE	FUNCIÓN
ARDUINO UNO R3 (x2)	ADQUISICIÓN DE EMG Y CONTROL DE SERVOMOTORES
SENSOR MYOWARE 2.0	DETECCIÓN DE SEÑALES MUSCULARES
SHIELD MYOWARE PARA ARDUINO	INTERFAZ SEGURA ENTRE SENSOR Y MICROCONTROLADOR
PCA9685	GENERACIÓN DE PWM PARA SERVOMOTORES
SERVOMOTORES MG995 (x5)	MOVIMIENTO DE LOS DEDOS DE LA PRÓTESIS
FUENTE DE ALIMENTACIÓN 5V/3A	ENERGÍA PARA LOS MOTORES

B. Flujo del sistema de adquisición y entrenamiento del modelo

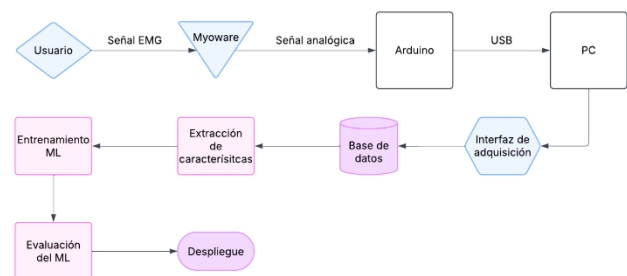


Fig. 2. Flujo del sistema

La Fig. 2 presenta el flujo completo del sistema hasta su integración en el sistema de control final. El proceso inicia

con el usuario, quien genera señales EMG a través de la contracción del músculo del antebrazo. Estas señales son captadas por el sensor MyoWare 2.0 entregando una señal analógica lista para ser muestreada.

Posteriormente, el Arduino Uno R3 digitaliza la señal EMG y la envía a través de USB a una computadora. En la PC, una interfaz de adquisición desarrollada en Python permite registrar la identidad del usuario, el número de sesión, y guía visualmente al usuario en la ejecución de los gestos requeridos (abierto, cerrado o reposo).

La interfaz almacena los datos adquiridos como archivos .csv, los cuales se organizan en una base de datos interna. Desde esta base, se utilizan los datos para una fase de extracción de características, donde se calcula el valor absoluto medio, la desviación estándar y el valor máximo para cada ventana de señal.

Estas características son empleadas como entradas para el entrenamiento de modelos de Machine Learning. En este proyecto, se evaluaron tres algoritmos: Árbol de decisión, Random Forest y SVM. Finalmente, el modelo entrenado fue desplegado para su uso en tiempo real dentro del sistema de control de la prótesis.

C. Adquisición de datos

La señal electromiográfica fue obtenida utilizando el sensor MyoWare 2.0 Muscle Sensor Development Kit, el cual se colocó sobre el músculo Flexor Digitorum Profundus del antebrazo para captar la actividad eléctrica superficial asociada al movimiento de la mano.



Fig. 3. Músculo Flexor Digitorum Profundus

Este músculo fue seleccionado debido a que presenta una diferenciación clara y consistente en su señal electromiográfica durante las acciones de apertura, cierre y reposo de la mano, lo que lo convierte en una opción óptima para tareas de clasificación basadas en gestos musculares.

El sensor utilizado incluye internamente un sistema de amplificación, un filtro pasa-bandas (20–450 Hz) y un filtro notch a 60 Hz, entregando una señal analógica acondicionada, lista para su muestreo por microcontroladores. En la Fig. 4 se muestra el circuito interno del sensor, destacando la etapa de filtrado y acondicionamiento de señal que permite su uso inmediato con sistemas embebidos.

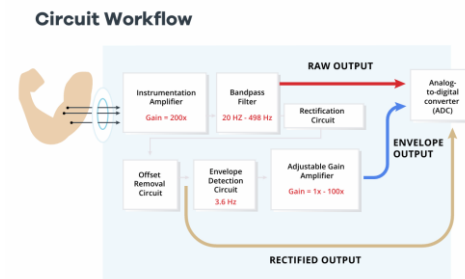


Fig. 4. Circuito interno del sensor MyoWare 2.0. Imagen tomada de [6].

Para la adquisición, el sensor se conectó a un shield de Arduino incluido en el kit, el cual se montó sobre un Arduino Uno R3. Este microcontrolador enviaba los datos en tiempo real a una computadora mediante conexión USB. En la computadora, se desarrolló un script en Python con una interfaz gráfica que permitía registrar el ID del usuario, el número de sesión e indicar el gesto a realizar. La interfaz guiaba visualmente al usuario, indicando en pantalla cuándo debía ejecutar cada gesto muscular (abierto, cerrado o en reposo), lo que permitió estandarizar el proceso de captura.

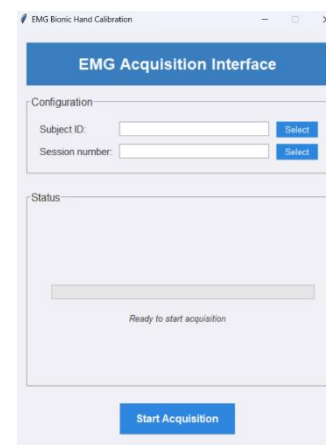


Fig. 5. Interfaz de adquisición de datos

Cada vez que se capturaba una secuencia, el sistema generaba automáticamente un archivo .csv etiquetado con el ID del usuario, el número de sesión y los gestos incluidos. Cada archivo .csv contenía, además de las muestras de señal EMG, la etiqueta correspondiente al gesto muscular ejecutado, codificada numéricamente como: reposo (0), mano abierta (1) y mano cerrada (2). Por cada usuario se realizaron aproximadamente cinco sesiones, logrando recolectar un total de 40 capturas distintas. Cada sesión se estructuró en siete intervalos consecutivos en el siguiente orden: reposo, abierto, reposo, cerrado, reposo, abierto, cerrado. En total, cada archivo contenía alrededor de 700 muestras en total, registradas con una frecuencia adecuada para capturar los cambios dinámicos en la actividad muscular. La Fig. 6 muestra una gráfica real correspondiente a una de estas sesiones de captura.

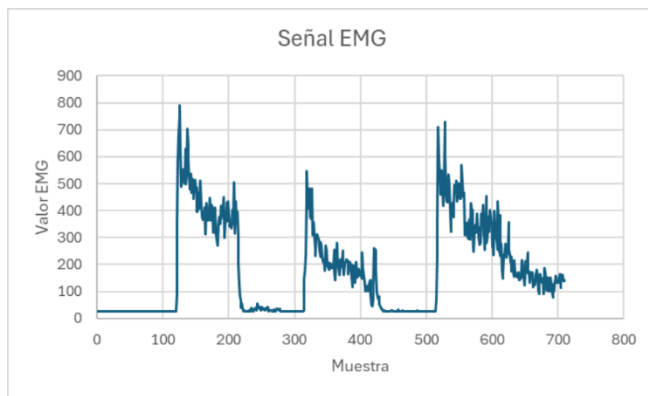


Fig. 6. Señal EMG en reposo, abierto y cerrado

D. Procesamiento y extracción de características

En la computadora cliente, se implementó un script en Python que procesa ventanas de señal en tiempo real. Para cada ventana, se calculan tres características estadísticas del dominio temporal: valor absoluto medio (Mean Value), desviación estándar (Standard Deviation) y valor máximo (Maximum Value). Estas características fueron seleccionadas por su bajo costo computacional y buena capacidad de discriminación entre contracciones musculares.

El sistema fue optimizado para operar con ventanas temporales que permiten un compromiso entre estabilidad de la predicción y velocidad de respuesta, lo cual es crucial para el uso en tiempo real.

E. Entrenamiento del modelo

Los modelos fueron entrenados utilizando los vectores de características extraídas como entrada, y las clases codificadas como variable objetivo: 0 para reposo, 1 para mano abierta y 2 para mano cerrada, según el etiquetado aplicado durante la adquisición.

El conjunto de datos fue cargado desde un archivo .csv, y se dividió en dos subconjuntos: 90 % para entrenamiento y 10 % para prueba, utilizando una semilla fija para garantizar reproducibilidad. Las características utilizadas fueron: valor absoluto medio (Mean value), desviación estándar (Std Dev) y valor máximo (Maximum).

Se entrenaron tres modelos de clasificación supervisada: Árbol de decisión, Máquina de vectores de soporte (SVM) y Random Forest, todos implementados con la biblioteca scikit-learn. Una vez entrenados, se evaluó su rendimiento usando la métrica de precisión (accuracy) sobre los datos de prueba. Los resultados obtenidos fueron los siguientes: El Árbol de Decisión obtuvo una precisión del 80 %, la SVM del 67 %, y el Random Forest del 83 %.

El modelo con mayor precisión fue seleccionado automáticamente, y posteriormente se guardó en un archivo .pkl utilizando la biblioteca joblib para su despliegue en el sistema. Adicionalmente, se generó una matriz de confusión para evaluar el rendimiento por clase y un gráfico de comparación entre etiquetas reales y predichas para visualizar la precisión del modelo en muestras individuales.

F. Diseño mecánico y construcción de la prótesis

El diseño mecánico de la prótesis se centró en lograr una estructura funcional, ligera y fácilmente replicable, priorizando la capacidad de realizar movimientos naturales de apertura y cierre de la mano. La prótesis consta de cinco dedos, incluyendo el pulgar, y fue construida de forma que cada dedo pudiera ser controlado de manera independiente por un servomotor individual. Esta configuración permite no solo los gestos de abrir y cerrar, sino también posiciones intermedias continuas, gracias al uso de un sistema de cables tipo tendón que transmiten la fuerza desde los motores hacia las falanges.

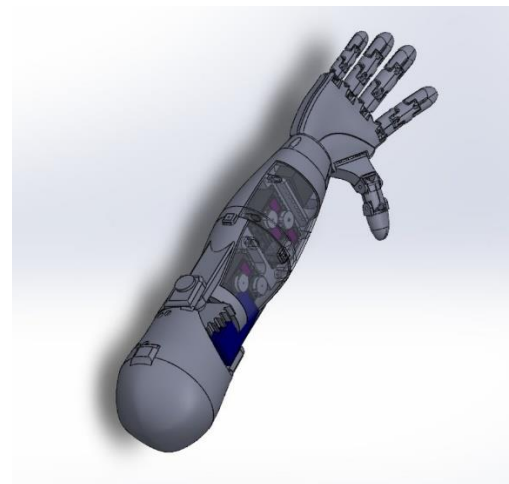


Fig. 7. Diseño 3D en Solidworks con vista a los motores

El modelo fue adaptado a partir de un diseño base en SolidWorks, con modificaciones enfocadas en optimizar su comportamiento mecánico y compatibilidad con servomotores estándar. La forma final responde exclusivamente a criterios de funcionalidad, buscando un compromiso entre rigidez estructural, movilidad y facilidad de ensamblaje.

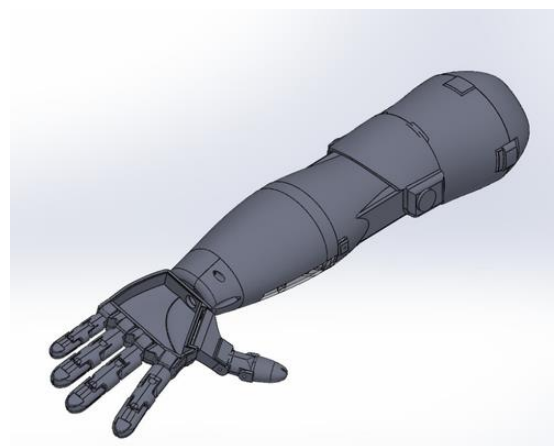


Fig. 8. Diseño 3D de la prótesis

La prótesis fue fabricada mediante impresión 3D FDM, utilizando filamento PLA, elegido por su disponibilidad, bajo costo y facilidad de uso. Las piezas se unieron mediante tornillos M2, lo cual permite desmontajes parciales para mantenimiento o ajustes. La instalación de los servomotores se realizó dentro de una base fija, desde la cual se enrutan los hilos hacia las articulaciones de cada dedo.

Tabla II. COMPONENTES MECÁNICOS DE LA PRÓTESIS

COMPONENTE	DESCRIPCIÓN
PLA	MATERIAL PARA IMPRESIÓN 3D
TORNILLOS M2	ENSAMBLE DE PARTES MÓBILES
LÍNEA DE PESCA	TRANSMISIÓN DE FUERZA DESDE LOS SERVOS
SERVOMOTORES	CONTROL INDEPENDIENTE PARA CADA DEDO

IV. RESULTADOS

A. Precisión del modelo y rendimiento de clasificación

El modelo seleccionado para el sistema fue Random Forest, el cual alcanzó una precisión del 83 % sobre el conjunto de prueba. Para validar el rendimiento por clase, se utilizó una matriz de confusión (Fig. 9), que muestra el siguiente comportamiento:

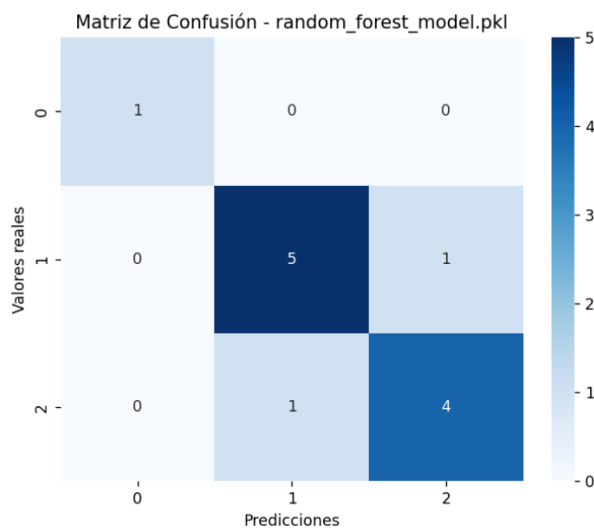


Fig. 9. Matriz de confusión Random Forest

- La clase 0 (*reposo*) fue clasificada correctamente en 1 de 1 caso ($n = 1$).
- La clase 1 (*abierto*) fue clasificada correctamente en 5 de 6 casos ($n = 6$), mientras que en 1 caso fue clasificada erróneamente como cerrado.
- La clase 2 (*cerrado*) fue clasificada correctamente en 4 de 5 casos ($n = 5$), presentándose una clasificación errónea como abierto.

Estos resultados reflejan una buena capacidad discriminativa general del modelo, aunque se observa cierta confusión entre las clases 1 y 2, es decir, entre abierto y cerrado, lo cual es consistente con la naturaleza de las señales EMG cuando las contracciones no son suficientemente diferenciadas.

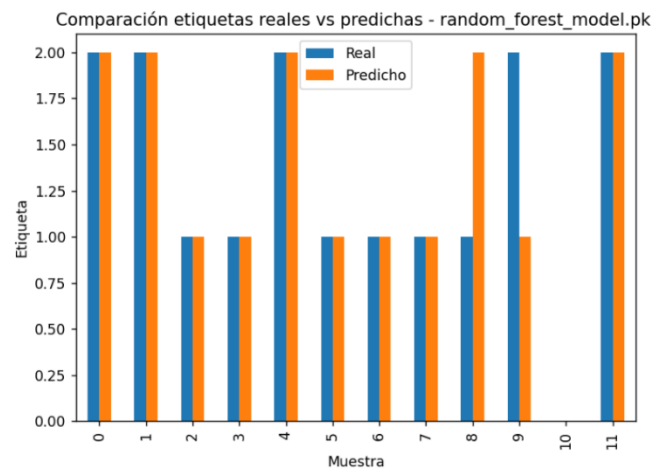


Fig. 10. Comparación entre etiquetas reales y predichas

La Fig. 10 muestra una comparación directa entre las etiquetas reales y las etiquetas predichas para un subconjunto de 12 muestras. Se puede observar que, aunque la mayoría de las predicciones coinciden con las etiquetas reales correspondientes, se presentan algunos errores puntuales, particularmente en las muestras intermedias donde abierto fue interpretado como cerrado.

B. Validación con usuarios

Para validar el sistema en condiciones reales, se realizaron pruebas con cinco usuarios sanos. Cada usuario fue instruido para ejecutar los tres gestos musculares definidos: reposo, abierto y cerrado. El sensor fue colocado sobre el músculo Flexor Digitorum Profundus, siguiendo el mismo protocolo de adquisición utilizado durante el entrenamiento.

Durante las pruebas, el sistema fue capaz de detectar correctamente los tres estados musculares en todos los participantes, replicando los resultados obtenidos en entrenamiento. El modelo mostró un rendimiento estable, sin necesidad de recalibración por usuario.

C. Tiempo de respuesta

El sistema demostró un tiempo de respuesta promedio de aproximadamente un segundo, desde la activación muscular hasta la ejecución completa del movimiento de la prótesis. Este tiempo incluye:

- Captura de señal EMG.
- Transmisión serial hacia la PC cliente.
- Ejecución del modelo de clasificación.
- Comunicación mediante TCP/IP hacia el servidor.
- Activación de servomotores vía Arduino y PCA9685.

Este rendimiento permite su uso en aplicaciones funcionales de agarre básico.

D. Evaluación física de la prótesis



Fig. 11. Prótesis construida

La prótesis fue construida mediante impresión 3D FDM utilizando filamento PLA, material que proporcionó una buena relación entre rigidez, ligereza y facilidad de fabricación. Se emplearon tornillos M2 para el ensamblaje de las partes móviles y un sistema de cables tipo tendón para conectar los servomotores con las falanges de cada dedo.

Durante las pruebas prácticas, el sistema mecánico demostró ser robusto y funcional, permitiendo la apertura y cierre de los cinco dedos sin atasco mecánico ni deformación estructural. El diseño fue lo suficientemente compacto y modular como para permitir ajustes rápidos, lo que favorece futuras iteraciones del prototipo.

El costo total estimado del sistema, incluyendo los componentes electrónicos, el sensor EMG y los materiales de impresión 3D, fue de \$4,747.25 MXN, véase la Tabla III en el Apéndice A.

E. Validación visual y demostrativa

Se elaboró un video demostrativo del funcionamiento del sistema en tiempo real, donde se documenta a un usuario ejecutando los tres gestos y la respuesta inmediata de la prótesis. El video demostrativo se encuentra disponible en el repositorio oficial del proyecto, véase referencia [5].

V. CONCLUSIONES Y TRABAJO A FUTURO

A. Conclusiones

Este proyecto logró demostrar la viabilidad técnica de una prótesis de mano biónica controlada mediante señales electromiográficas (EMG) y clasificada en tiempo real utilizando modelos de Machine Learning. Se diseñó y construyó una prótesis funcional de cinco dedos con servomotores independientes, capaz de ejecutar tres gestos: reposo, mano abierta y mano cerrada.

El sistema fue validado con cinco usuarios sanos, mostrando una precisión del 83 % con el modelo Random Forest, y un tiempo de respuesta de un segundo, incluyendo todo el flujo desde la contracción muscular hasta el movimiento mecánico. La arquitectura distribuida basada en TCP/IP permitió separar físicamente el sistema de lectura y el de control, resolviendo problemas de interferencia eléctrica en el sensor EMG.

Además, se desarrolló un sistema de adquisición de datos robusto, con interfaz gráfica, etiquetado automático y generación de archivos CSV. Este pipeline permitió entrenar múltiples modelos de clasificación y seleccionar el mejor para su integración al sistema final.

B. Áreas de mejora

Si bien el sistema funcionó correctamente, se identificaron errores esporádicos al clasificar el gesto de mano abierta, el cual fue en ocasiones interpretado como cerrado. Esto podría mitigarse mediante:

- Aumento de la base de datos con más usuarios y contracciones de diferente intensidad y duración.
- Mejor calibración de la extracción de características o inclusión de características adicionales.
- Posible filtrado dinámico o segmentación adaptativa de las ventanas EMG.

C. Trabajo futuro

El siguiente desarrollo del sistema se enfocará en convertir el prototipo actual en una solución completamente portátil, autónoma y con mayor capacidad funcional:

- **Despliegue embebido del modelo de clasificación:** Se realizará la conversión del modelo de Machine Learning a formato **TensorFlow Lite**, para su ejecución en un microcontrolador embebido. Esto eliminará la dependencia de una PC externa y permitirá procesamiento local en tiempo real, optimizando el consumo energético y la portabilidad.
- **Integración completa en la prótesis física:** Todos los componentes —sensores EMG, controlador PCA9685, microcontrolador y fuente de alimentación— serán incorporados dentro del cuerpo de la prótesis. Esto permitirá un sistema autónomo y listo para uso en campo, sin cables externos ni módulos separados.
- **Expansión del sistema de entrada EMG:** Se añadirán múltiples sensores EMG distribuidos en distintos músculos del antebrazo. Esta mejora permitirá una clasificación más precisa y la detección de un conjunto ampliado de gestos musculares, aumentando así la funcionalidad y versatilidad de la prótesis.
- **Implementación de control híbrido EEG-EMG:** Se integrarán sensores EEG para identificar la intención del gesto (por ejemplo, abrir, cerrar, pinzar), mientras que las señales EMG se utilizarán para ejecutar el gesto mediante un control proporcional del movimiento. Este enfoque híbrido mejorará la intuitividad del sistema y reducirá la carga física del usuario durante el uso prolongado.

REFERENCIAS

- [1] Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32. <https://doi.org/10.1023/A:1010933404324>
- [2] Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3), 273–297. <https://doi.org/10.1007/BF00994018>
- [3] De Luca, C. J. (2006). Electromyography. In *Encyclopedia of Medical Devices and Instrumentation* (2nd ed., Vol. 2, pp. 98–109). Wiley. <https://doi.org/10.1002/0471732877.emd078>
- [4] Hudgins, B., Parker, P., & Scott, R. N. (1993). A new strategy for multifunction myoelectric control. *IEEE Transactions on Biomedical Engineering*, 40(1), 82–94. <https://doi.org/10.1109/10.204774>
- [5] Pérez, I. (2025). *Repositorio del proyecto de prótesis biónica controlada por EMG*. GitHub. <https://github.com/isaiperezhr/Bionic-Hand>
- [6] *MyoWare 2.0 Advanced Guide*, SparkFun Electronics, 2022. [Online]. Disponible: https://myoware.com/wp-content/uploads/2022/03/MyoWare_v2_AdvancedGuide-Updated.pdf
- [7] Phinyomark, A., Limsakul, C., & Phukpattaranont, P. (2009). A novel feature extraction for robust EMG pattern recognition. *Journal of Computing*, 1(1), 71–80
- [8] Sharma, A., & Badlani, N. (2019). Review on EMG signal analysis and classification techniques. In *2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU)* (pp. 1–5). IEEE. <https://doi.org/10.1109/IoT-SIU.2019.8777554>

APPENDIX A – Costos del sistema

La siguiente tabla presenta un desglose estimado de los costos asociados a los componentes eléctricos, electrónicos y materiales utilizados en la fabricación de la prótesis biónica.

Tabla III. Costos estimados del sistema

Componente	Cantidad	Costo Unitario (MXN)	Costo total (MXN)
Arduino Uno R3	2	\$180	\$360
MyoWare 2.0 Muscle Sensor Development Kit	1	\$3,298.90	\$3,298.90
PCA9685 Servo Driver	1	\$136.12	\$136.12
Servomotores MG995 Kit	1	\$352.23	\$352.23
Impresión 3D PLA	-	-	\$500.00
Tornillos y cableado	-	-	\$100.00
Total	-	-	\$4,747.25