

**ESTUDIO COMPARATIVO DE TÉCNICAS DE MACHINE LEARNING  
PARA EL CONTROL DE MOVIMIENTO DE UNA PRÓTESIS DE  
EXTREMIDAD SUPERIOR**



**LAURA DANIELA LASSO ARCINIEGAS  
BRAYAN ANDRES VIVEROS MELO**



**UNIVERSIDAD DE NARIÑO  
FACULTAD DE INGENIERÍA  
DEPARTAMENTO DE ELECTRÓNICA  
SAN JUAN DE PASTO**

**2019**

**ESTUDIO COMPARATIVO DE TÉCNICAS DE MACHINE LEARNING  
PARA EL CONTROL DE MOVIMIENTO DE UNA PRÓTESIS DE  
EXTREMIDAD SUPERIOR**

**LAURA DANIELA LASSO ARCINIEGAS  
BRAYAN ANDRES VIVEROS MELO**

**Trabajo de grado presentado como requisito para optar al título de  
Ingeniero Electrónico**

**Asesor**

**PhD. DIEGO HERNÁN PELUFFO ORDÓÑEZ**

**UNIVERSIDAD DE NARIÑO  
FACULTAD DE INGENIERÍA  
DEPARTAMENTO DE ELECTRÓNICA  
SAN JUAN DE PASTO**

**2019**

## **RESPONSABILIDAD**

”Las ideas y conclusiones aportadas en el siguiente trabajo son responsabilidad exclusiva del autor”. Artículo 1ro del Acuerdo No. 324 de octubre 11 de 1966 emanado del Honorable Consejo Directivo de la Universidad de Nariño.

Nota de Aceptación:

---

---

---

---

---

---

---

---

---

Firma del Director del Trabajo de Grado

---

Firma del Jurado

---

Firma del Jurado

San Juan de Pasto, Marzo de 2019

## AGRADECIMIENTOS

*Agradezco a Dios por las bendiciones recibidas. A mi madre, por todo su apoyo para seguir adelante y por su cariño durante este proceso. A mis hermanas por sus consejos y su ayuda. Finalmente a mis profesores por el conocimiento brindado y guía a lo largo de toda la carrera.*

Brayan Andres Viveros Melo

*Agradezco a mis padres y a mi tia por su apoyo y amor incondicional, a mis hermanos por su ejemplo, a mis sobrinos por ser mi motivación, a mis amigos y primos por acompañarme en cada etapa y finalmente, a mis profesores, que con su conocimiento y guía, permitieron culminar este trabajo con éxito.*

Laura Daniela Lasso Arciniegas

A Gloria, Raul y Ligia.

A Olga, Diana y Hely.

## RESUMEN

Las señales electromiográficas (EMG) son impulsos eléctricos producidas por los músculos durante los procesos de contracción y relajación. Esta característica fisiológica permite analizar el estado fisiológico de los músculos y además permite crear interfaces de control para comunicación entre usuario y máquina. Entre los diferentes tipos de aplicaciones de interfaces han tomado auge aquellas dedicadas a individuos con discapacidades motrices que dificulten la realización de actividades cotidianas.

El desarrollo de tecnologías de control basadas en señales EMG requiere de sistemas que interpreten las órdenes del usuario. Particularmente, para el control de prótesis es necesario llevar a cabo un proceso de interpretación semántica de las señales EMG para ejecutar un determinado movimiento, este proceso puede realizarse mediante técnicas supervisadas de *machine learning*. No obstante, pese a que se han desarrollado diversos estudios relacionados y en vista de que no todos los métodos garantizan resultados óptimos, aún sigue siendo un problema abierto el desarrollo de un criterio universal para comparar técnicas y determinar cuáles son las más adecuadas para las tareas de control de prótesis de extremidad superior.

En este trabajo se realiza un estudio comparativo de técnicas supervisadas de machine learning y caracterización de señales EMG para el reconocimiento de movimientos básicos de la mano, aplicado al control de una prótesis de extremidad superior, con el fin de determinar las técnicas con mayor precisión para tareas de bio-rehabilitación.

**PALABRAS CLAVE:** Red Neuronal Artificial, Reducción de dimensión, Señal EMG, KNN, Parzen, Wavelet.

# ABSTRACT

Electromyographic signals (EMG) are electrical impulses produced by muscles during the processes of contraction and relaxation. This physiological feature of the muscles allows to create control interfaces for communication between user and machine. Among the different types of applications, some interfaces are dedicated to individuals with motor disabilities that difficult to perform daily activities.

The development of control technologies based on EMG signals requires systems that interpret the user's commands. In particular, for the control of prostheses it is necessary to carry out a process of semantic interpretation of the EMG signals to execute a certain movement, this process can be carried out through supervised machine learning techniques. However, despite the fact that several related studies have been developed and since not all methods guarantee optimal results, it still remains as an open problem to develop a universal criterion to compare techniques and determine which are the most appropriate for the tasks of upper limb prosthesis control.

In this work, we performed the comparative study of supervised machine learning techniques and characterization of EMG signals for the recognition of basic movements of the hand, applied to the control of a prosthesis of the upper limb, in order to determine the techniques with greater efficiency for bio-rehabilitation tasks.

**KEYWORDS:** Artificial Neural Network, Dimension reduction, EMG signal, KNN, Parzen, Wavelet.



# TABLA DE CONTENIDO

	pág.
<b>INTRODUCCIÓN . . . . .</b>	<b>19</b>
Planteamiento del problema . . . . .	20
Justificación . . . . .	21
Contribución de esta tesis . . . . .	22
Objetivos . . . . .	23
Objetivo General . . . . .	23
Objetivos Específicos . . . . .	23
Marco Teórico . . . . .	24
Anatomía muscular miembro superior . . . . .	24
Equipo de electromiografía . . . . .	26
Procesamiento de Señales . . . . .	29
Caracterización. . . . .	30
Selección de atributos . . . . .	32
Aprendizaje de máquina . . . . .	35
<b>1. METODOLOGÍA . . . . .</b>	<b>42</b>
1.1 Base de Datos . . . . .	42
1.2 Adquisición de señales . . . . .	46
1.3 Pre-procesamiento . . . . .	47
1.4 Caracterización . . . . .	48
1.5 Selección de movimientos . . . . .	51
1.6 Selección de características . . . . .	52
1.7 Clasificación . . . . .	54

1.8 Visualización . . . . .	54
1.8.1 Inicio. . . . .	54
1.8.2 Entrenamiento. . . . .	55
1.8.3 Prueba de los modelos. . . . .	56
1.9 Actuador . . . . .	58
<b>2. MARCO EXPERIMENTAL . . . . .</b>	<b>62</b>
2.1 Medidas de desempeño . . . . .	62
2.2 Pruebas aplicadas a la base de datos. . . . .	63
2.3 Prueba de Interfaz gráfica y Actuador . . . . .	66
2.4 Pruebas realizadas a las señales adquiridas con la interfaz de usuario . . . . .	66
<b>3. RESULTADOS Y ANÁLISIS . . . . .</b>	<b>68</b>
3.1 Resultados de la identificación del conjunto de características . . . . .	68
3.2 Resultados de clasificación de señales EMG de Ninapro . . . . .	73
3.3 Resultados de la Interfaz gráfica y Actuador . . . . .	87
3.4 Resultados de la prueba de modelos KNN y ANN con señales adquiridas . . . . .	92
<b>4. CONCLUSIONES . . . . .</b>	<b>99</b>
<b>5. TRABAJO FUTURO . . . . .</b>	<b>101</b>
<b>BIBLIOGRAFÍA . . . . .</b>	<b>102</b>
<b>ANEXOS . . . . .</b>	<b>105</b>

# LISTA DE FIGURAS

	pág.
<b>Figura 1</b> Electrodo para electromiografía invasiva tipo aguja concéntrica desechable. Longitud de 3 - 4 cm de acero, con plástico o barniz aislante excepto en la punta. . . . .	27
<b>Figura 2</b> Descomposición Wavelet en 3 niveles. El cuadro gris representa la señal original, en el verde se representan los coeficientes de aproximación consecuentes del nivel anterior; finalmente en el cuadro amarillo los correspondientes coeficientes de detalle. . . . .	33
<b>Figura 3</b> Modelo de clasificador KNN, donde la nueva instancia en color verde se clasifica como perteneciente a la clase 1, debido a que en sus 3 vecinos más cercanos la mayoría son muestras de la clase 1. . . . .	37
<b>Figura 4</b> Modelo de perceptrón, donde la salida $u$ se obtiene al aplicar la función sigmoïdal a la sumatoria de las entradas $x_{i,j}$ multiplicadas por los pesos $w_i$ . . . . .	38
<b>Figura 5</b> Metodología para el estudio comparativo de técnicas <i>Machine Learning</i> . Los procesos que se compartieron entre la base de datos y toma de señales se rellenaron con color gris; los procesos de modelamiento se rellenaron con color naranja y los de aplicación en color azul. . . . .	43
<b>Figura 6</b> Movimientos base de datos Ninapro. Coloreados de azul los movimientos de dedos; en rosa 9 movimientos de muñeca y 8 configuraciones isométricas e isotónicas; y en verde 23 movimientos de agarre . . . . .	44
<b>Figura 7</b> Protocolo toma de señales base de datos. Se muestra a un sujeto con electrodos conectados en su antebrazo, siguiendo un vídeo en la pantalla y ejecutando movimientos mientras se registran las señales. . . . .	46
<b>Figura 8</b> Sensor muscular Myoware. Mide la actividad de un músculo al monitorear el potencial eléctrico generado por las células musculares. . . . .	47

<b>Figura 9</b>	Los 3 movimientos que ejecuta el usuario,a) Movimiento 1: Agarre pequeño, b)Movimiento 2: Agarre lateral y c)Movimiento 3: Agarre de tripode de escritura. . . . .	48
<b>Figura 10</b>	Posicionamiento de los electrodos alrededor del antebrazo para toma de señales y posterior entrenamiento de clasificadores. . . . .	49
<b>Figura 11</b>	Movimientos de la base de datos que presentaron mayor separabilidad.	53
<b>Figura 12</b>	Vista pantalla principal de la interfaz para el entrenamiento de clasificadores. . . . .	55
<b>Figura 13</b>	Etapas de inicio de la interfaz gráfica. . . . .	56
<b>Figura 14</b>	Etapas de Entrenamiento en la interfaz gráfica. Se recortaron las señales EMG adquiridas, se caracterizaron y se entrenaron los clasificadores KNN Y ANN. . . . .	57
<b>Figura 15</b>	Etapas de prueba de modelo en la interfaz gráfica, tomando una nueva señal y realizando predicción con modelos entrenados. . . . .	58
<b>Figura 16</b>	Modelo 3D de las partes del brazo robótico, a) Antebrazo, b) Soporte para servomotores, c) Accesorio para servomotores y d)Unión Palma de la mano y Antebrazo. . . . .	59
<b>Figura 17</b>	Modelo 3D de las partes del brazo robótico implementado, a)Palma de la mano del brazo robótico, b)Dedos indice, medio, anular y meñique, c)Acople dedo pulgar con servomotor y d) Acople servomotor con Antebrazo para giro. . . . .	60
<b>Figura 18</b>	Brazo robótico impreso en 3D y completamente armado. . . . .	61
<b>Figura 19</b>	Matriz de confusión. . . . .	63
<b>Figura 20</b>	Histograma del vector de atributos obtenido con Cfs Subset Eval. . . . .	69
<b>Figura 21</b>	Histograma del vector de atributos obtenido con Classifier Attribute Eval. . . . .	69
<b>Figura 22</b>	Histograma de vector de atributos arrojado por Correlation Attribute Eval. . . . .	70
<b>Figura 23</b>	Histograma del vector de características arrojado por Gain Ratio Attribute Eval. . . . .	71

<b>Figura 24</b>	Histograma del vector de características arrojado por Relief. . . .	72
<b>Figura 25</b>	Curvas de error de clasificación promedio del clasificador KNN, variando el número de características, para cada uno de los métodos de selección de atributos. . . . .	76
<b>Figura 26</b>	Matriz de confusión de los mejores modelos de clasificador de KNN, para los diferentes métodos de selección de características. . . . .	77
<b>Figura 27</b>	Tablas de sensibilidad y especificidad promedio obtenidas de los experimentos: 6 - 10. . . . .	78
<b>Figura 28</b>	Curvas de error de clasificación promedio del clasificador probabilístico de Parzen, variando el número de características, para cada uno de los métodos de selección de atributos. . . . .	80
<b>Figura 29</b>	Matriz de confusión de los mejores modelos de clasificador probabilístico de Parzen, para los diferentes métodos de selección de atributos. . . . .	81
<b>Figura 30</b>	Tablas de sensibilidad y especificidad promedio obtenidas de los experimentos: 11 - 15. . . . .	82
<b>Figura 31</b>	Curvas de error de clasificación promedio de la Red Neuronal Artificial con Backpropagation, variando el número de características, para cada uno de los métodos de selección de atributos. . . . .	84
<b>Figura 32</b>	Matriz de confusión de los mejores modelos de la Red Neuronal Artificial con Backpropagation, para los diferentes métodos de selección de características. . . . .	85
<b>Figura 33</b>	Tablas de sensibilidad y especificidad promedio obtenidas de los experimentos: 16 - 20. . . . .	86
<b>Figura 34</b>	Resumen de los resultados . . . . .	87
<b>Figura 35</b>	Ejecución de la etapa de <i>Inicio</i> de la interfaz gráfica. . . . .	88
<b>Figura 36</b>	Ejecución de la etapa de <i>Entrenamiento</i> de la interfaz gráfica. . . . .	89
<b>Figura 37</b>	Ejecución de la etapa de <i>Prueba de los modelos</i> de la interfaz gráfica, movimiento 1. . . . .	89

<b>Figura 38</b>	Ejecución de la etapa de <i>Prueba de los modelos</i> de la interfaz gráfica, movimiento 2. . . . .	90
<b>Figura 39</b>	Ejecución de la etapa de <i>Prueba de los modelos</i> de la interfaz gráfica, movimiento 3. . . . .	90
<b>Figura 40</b>	Movimiento 1 ejecutado por el prototipo de prótesis. . . . .	91
<b>Figura 41</b>	Movimiento 2 ejecutado por el prototipo de prótesis. . . . .	91
<b>Figura 42</b>	Movimiento 3 ejecutado por el prototipo de prótesis. . . . .	92
<b>Figura 43</b>	Señales EMG del sujeto 1 realizando los 3 movimientos para prueba de modelos KNN y ANN con señales adquiridas . . . . .	94
<b>Figura 44</b>	Movimientos representado por las 3 mejores características del sujeto 1 para prueba de modelos KNN y ANN con señales adquiridas . . . . .	95
<b>Figura 45</b>	Señales EMG del sujeto 2 realizando los 3 movimientos para prueba de modelos KNN y ANN con señales adquiridas . . . . .	97
<b>Figura 46</b>	Movimientos representado por las 3 mejores características del sujeto 2 para prueba de modelos KNN y ANN con señales adquiridas . . . . .	98
<b>Figura 47</b>	Página web "Classification for movement identification from EMG signals". . . . .	135

## LISTA DE TABLAS

	<b>pág.</b>
<b>Tabla 1</b> Formulas utilizadas para caracterización . . . . .	51
<b>Tabla 2</b> Matriz de confusión de la prueba de modelo KNN con sujeto 1 . . .	93
<b>Tabla 3</b> Matriz de confusión de la prueba de modelo ANN con sujeto 1 . . .	93
<b>Tabla 4</b> Matriz de confusión de la prueba de modelo KNN con sujeto 2. . .	96
<b>Tabla 5</b> Matriz de confusión de la prueba de modelo ANN con sujeto 2. . .	96

## LISTA DE ANEXOS

	<b>pág.</b>
<b>ANEXO A</b> Artículo publicado en CCC 2018: Advances in Computing . . .	106
<b>ANEXO B</b> Artículo publicado en IWAIPR 2018: Progress in Artificial In- telligence and Pattern Recognition . . . . .	119
<b>ANEXO C</b> CERTIFICADO CONGRESO 13 CCC . . . . .	132
<b>ANEXO D</b> Página web EMG. . . . .	134



## GLOSARIO

**ALGORITMO:** Es un conjunto de instrucciones o reglas bien definidas, que permiten llevar a cabo una actividad siguiendo los pasos mencionados, para así obtener una solución.

**INTELIGENCIA ARTIFICIAL:** Es una area multidisciplinaria, donde se combinan algoritmos con el propósito de crear máquinas que presenten las mismas capacidades que el ser humano como: Aprender y resolver problemas.

**MUESTRA:** Es un conjunto de valores tomados de una señal continua.

**FRECUENCIA DE MUESTREO:** Es el número de muestras por unidad de tiempo que se toman de una señal continua para producir una señal discreta. Se expresa en hercios (Hz, ciclos por segundo).

**CARACTERÍSTICAS:** Se refiere a los atributos que describen las muestras de un conjunto de datos o de una señal.

**CLASE:** Agrupación de objetos o datos que tiene características en común.

**INTERFAZ GRÁFICA DE USUARIO:** También llamada como GUI (Graphical User Interface), es un programa informático que utiliza un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles en dicho programa. Su principal objetivo es brindar un entorno visual sencillo al usuario y así permitir una mejor comunicación con el sistema operativo de una máquina o computador.

# INTRODUCCIÓN

**Modalidad:** Investigación      **Línea de investigación:** Tratamiento de señales.

La electromiografía (EMG) es un proceso de diagnóstico médico orientado a conocer el funcionamiento de los músculos y nervios que los controlan. Tradicionalmente, del procedimiento de EMG, se obtiene un registro gráfico de la actividad eléctrica producida por los músculos esqueléticos durante el movimiento de dicho tejido (Navarro, 2016 ). Para obtener las señales provenientes de los músculos, se precisa de electrodos, que pueden ser de tipo invasivo o no invasivo, es decir, por inserción dentro del músculo o adheridos a la superficie de la piel. En la actualidad, los no invasivos son los más utilizados debido a que no causan dolor al paciente y no se requiere de un entrenamiento específico para su colocación (Cifuentes, 2010). Un proceso de electromiografía requiere, además de los sensores, de métodos avanzados de procesamiento. Una vez adquiridas las señales de EMG pueden ser usadas en aplicaciones clínicas y biomédicas, facilitando el diagnóstico de enfermedades neuromusculares (Navarro, 2016), y últimamente, han tomado fuerza para el desarrollo de prótesis robóticas de extremidades superiores e inferiores (Pambudi, 2016) (Vazquez, 2018).

La técnica de adquirir señales de EMG a través de electrodos de superficie se conoce como electromiografía de superficie (*sEMG*). En los primeros años de investigaciones sobre estas señales, se realizaron avances en la comprensión del funcionamiento de los músculos y, a partir de entonces, muchas investigaciones surgieron con el fin de brindar tecnologías que sean capaces de ayudar a las personas con discapacidad, trayendo consigo la necesidad de mejorar los métodos de reconocimiento de patrones, para identificar adecuadamente los movimientos del usuario, apoyados en algoritmos de aprendizaje supervisado y no supervisado (Cifuentes, 2010)(Vazquez, 2018)(Podrug, 2015).

Pese al desarrollo de tecnologías de conductores, microprocesadores y amplificadores,

la adquisición, procesamiento y clasificación de señales de EMG sigue siendo un reto. Debido a la naturaleza de las señales, se ve comprometido el registro adquirido, pues la amplitud de éstas puede variar desde el orden de microvoltios hasta 10 mV, y su frecuencia se encuentra alrededor de 50 Hz (Romo, 2007). Además de los electrodos, son necesarios amplificadores de instrumentación que pueden ser susceptibles al ruido, afectando la calidad de la señal y, en consecuencia, la identificación del movimiento ejecutado, haciendo necesario una búsqueda de características que mejoren el desempeño del algoritmo, y a la vez, un clasificador que sea capaz de afrontar el desafío de controlar varias funciones de una prótesis que reemplace una extremidad humana, de manera fiable e intuitiva. La complejidad de este problema se relaciona directamente con la cantidad de movimientos que el clasificador pueda diferenciar con eficacia. En este sentido, el campo de *Machine Learning* (aprendizaje automático) ofrece técnicas, algoritmos y herramientas para realizar dicha tarea de clasificación de forma automática. En particular, las técnicas supervisadas han reportado buenos resultados (Arozi, 2016)(Negi, 2016) (Cene, 2016).

En este trabajo se presentan los resultados de un estudio comparativo de técnicas supervisadas de *Machine Learning*, técnicas de reducción de dimensión y de características para la clasificación de señales EMG, generadas durante el movimiento de músculos de extremidades superiores. También se evalúa el desempeño de estas técnicas en el control de una prótesis electromecánica.

## **Planteamiento del problema**

Actualmente, la capacidad de adquisición de señales EMG ha abierto muchas posibilidades de aplicaciones, entre las que se encuentra la rehabilitación por medio de prótesis robóticas que puedan ser útiles en la vida de personas con discapacidad, quienes hasta hace tres décadas contaban únicamente con diseños básicos y rudimentarios para poder reemplazar miembros amputados o disfuncionales.

A pesar del desarrollo de la tecnología de adquisición y procesado de señales de EMG, estas prótesis modernas necesitan además de un sistema inteligente que sea capaz de identificar de manera efectiva los movimientos que el usuario desee realizar, que sea eficiente computacionalmente y que permita el control de una prótesis, por tanto la identificación de algoritmos de caracterización y clasificación de señales EMG con costo computacional moderado es de gran interés científico.

## **Justificación**

Según el informe mundial sobre la discapacidad (2011) publicado por la Organización Mundial de la Salud, más de mil millones de personas sufren algún tipo de discapacidad. Solo en Colombia, más de un millón de personas se ven afectadas por algún tipo de discapacidad estando un 7% de ellas relacionadas con los miembros superiores. Se estima que alrededor del mundo casi 200 millones de personas experimentan dificultades considerables, dificultando su desarrollo en el ámbito personal, social y económico. Con señales EMG es posible desarrollar prótesis electromecánicas que emulen movimientos de una mano biológica colaborando en la rehabilitación de estos pacientes y generando una mejor calidad de vida para ellos y sus familias. El control de prótesis dependen de señales que deben ser clasificadas con eficiencia y rapidez para que la prótesis ejecute el movimiento identificado sin ningún retraso o margen de error. Este proceso puede ser llevado a cabo por medio de distintas técnicas de *Machine Learning*, sin embargo no existe un criterio que permita saber qué técnica tendrá resultados útiles y además sin un costo computacional elevado.

## Contribución de esta tesis

Este proyecto investigativo puede llegar a ser importante para aquellas personas que han sufrido algún tipo de discapacidad que les impida desarrollar sus actividades diarias, así como para la comunidad científica en general que busca criterios que permitan clasificar señales EMG con fines educativos y de aplicaciones técnicas. Para las partes ya mencionadas es de gran interés poder conocer una combinación adecuada de técnicas de clasificación, caracterización, canales de entrada y número de características necesarias para diseñar y construir sistemas que puedan ser controlados a partir de señales de electromiografía y que envíen las órdenes a elementos mecánicos.

Por lo tanto, es pertinente realizar un estudio comparativo de técnicas supervisadas de *Machine Learning* para la clasificación de señales EMG, donde se evalúa la eficiencia de dichas técnicas en el control de prótesis de extremidad, tomando como criterio la efectividad de clasificación.

Es importante destacar que el desarrollo de esta investigación cuenta con el respaldo del Grupo de Investigación en Ingeniería y Electrónica de la Universidad de Nariño (GIIEE) y Smart Data Analysis Systems Group (SDAS Group).

# OBJETIVOS

**Objetivo General** Realizar un estudio comparativo de técnicas supervisadas de *Machine Learning* con el fin de determinar las más adecuadas para realizar el control de movimiento de una prótesis de extremidad superior.

## Objetivos Específicos

- Comparar técnicas supervisadas de *Machine Learning* aplicadas a características de señales EMG para determinar aquellas que presentan mejores índices de clasificación de movimientos de extremidad superior.
- Implementar una prótesis mecánica de extremidad superior que emule los movimientos básicos que realiza la mano humana a partir de diseños recomendados por la comunidad científica.
- Desarrollar un prototipo mecánico inteligente de prótesis de extremidad superior, que permita controlar los movimientos con señales EMG, con el fin de brindar una asistencia en el proceso de rehabilitación.

## MARCO TEÓRICO

En esta sección se dan a conocer diferentes aspectos que intervienen en el desarrollo de sistemas de control accionados por señales EMG. Se describen: Aspectos básicos de la biología involucrada, equipo electrónico de amplificación y adaptación, métodos de caracterización, selección de atributos y técnicas de clasificación.

**Anatomía muscular miembro superior** El miembro superior humano se caracteriza por su gran versatilidad de movimientos gracias a los músculos. Con la mano y antebrazo se realizan la mayor parte de las actividades humanas diarias como tomar monedas, manipular dispositivos, abrocharse una camisa, agarrar los cubiertos y demás acciones cotidianas. La eficiencia de las funciones de la mano se deben a la capacidad de combinaciones de posición de las articulaciones de la cintura escapular, el hombro, el codo, radioulnares y el carpo. El miembro superior está formado por cuatro segmentos principales:

- **Hombro:** Es el segmento proximal del miembro en el cual se superponen partes del tórax y dorso y de la porción lateral e inferior del cuello. Comprende las regiones pectoral, escapular y deltoidea del miembro superior y la parte lateral de la región cervical lateral (Moore, 2013).
- **Brazo:** Es el primer segmento del miembro superior y el de mayor movilidad. Se extiende desde el hombro hasta el codo y comprende las regiones anterior y posterior centradas en el húmero. Este segmento está compuesto por los siguientes músculos: *Coracobraquial*, que ayuda en la extensión y como depresor del hombro; *Braquial*, responsable de la flexión del codo; el *Bicep* es el encargado de la flexión del codo y supinación del antebrazo y, finalmente, el *Tricep* que realiza la función de extensión del codo (Moore, 2013).

- **Antebrazo:** Se extiende entre el codo y el carpo, conectando ambas articulaciones y comprende las regiones anterior y posterior del antebrazo, que recubren el radio y la ulna. Entre los músculos con mayor funcionalidad de este segmento se encuentra el  *cubital posterior, extensores de dedos y muñeca, abductor y extensor de pulgar e índice*. Esta sección del miembro superior tiene los músculos encargados de gran parte de los movimientos de la mano. (Moore, 2013).
- **Mano:** Parte del miembro superior distal del antebrazo que se estructura alrededor del carpo, el metacarpo y las falanges. Está cubierta con numerosas terminaciones sensibles al tacto, el dolor y la temperatura. La mano comprende el *carpo, la palma y el dorso de la mano y los dedos*(entre los que se cuenta un pulgar oponible) (Moore, 2013).

**Potenciales de acción.** El sistema nervioso somático además de conducir información sensitiva, se encarga de llevar información de control a los músculos esqueléticos. A través de impulsos eléctricos se logra el control de las extremidades. Esto se debe a que las membranas de las células excitables se encuentran polarizadas, siendo el interior negativo con respecto al exterior. Este potencial se origina por la concentración de diversos iones como ( $Na^+$ ,  $K^+$ ,  $Ca^{++}$  y  $Cl^-$ ) (Navarro, 2016).

En el músculo se puede distinguir la fibra muscular y la unidad motora. La fibra muscular se caracteriza por ser alargada, cilíndrica de unos 50  $\mu m$  de diámetro, con numerosos núcleos y tener en su citoplasma un material proteico contráctil que al ser estimulado se contrae desarrollando fuerza (Díaz, 2012). La activación de cada fibra muscular se hace a través del axón de la fibra nerviosa motriz que la inerva. Según la posición y la función del músculo, el número de fibras musculares inervadas por un mismo axón puede variar entre uno y más de mil. El conjunto formado por la célula nerviosa motriz en la espina dorsal, su axón y las fibras musculares que este inerva constituye la unidad funcional básica del sistema muscular y se conoce por unidad motora. (Navarro, 2016).



De esta manera puede considerarse al músculo como un conjunto de unidades motoras dispuestas en paralelo, con un tipo de fibras musculares modificadas llamadas *Husos musculares*, que perciben las tracciones en el músculo y sirven para el servocontrol de la posición del músculo. En la EMG se registra la actividad del músculo y se puede distinguir la activación de sus unidades motoras, las variaciones características de estas activaciones y las relaciones de unas unidades con otras.

**Equipo de electromiografía** En las siguientes subsecciones se describe los elementos de un equipo para registrar señales EMG.

**Electrodos.** Es un dispositivo conductor tipo placa o aguja pequeña que recoge la actividad eléctrica de los músculos a través de la piel o por inserción. Esta diferenciación de tipos de electrodos da lugar a la clasificación de electromiografía tipo *Invasiva* y *No invasiva*.

**Electromiografía invasiva:** Se utiliza para realizar el registro del potencial generado por una única unidad motora. Este tipo de electromiografía utiliza electrodos de inserción *Monopolares* o *Coaxiales*, los primeros consisten en una aguja corriente cuya longitud total ha sido aislada, la diferencia de potencial se mide en el extremo que se inserta en el músculo y el electrodo de referencia ubicado en la piel. Los de tipo *Coaxial*, consiste en una aguja hipodérmica a través de cuyo interior se han insertado uno o varios conductores metálicos finos aislados entre sí y con respecto a la aguja, sólo el extremo de estos conductores se encuentra desprovisto de aislamiento y es por este punto por el que se captura la señal procedente del tejido muscular.

Debido a que la inserción de este tipo de electrodos es dolorosa y requiere supervisión médica, su uso se limita al diagnóstico y rehabilitación en medicina interna o trauma-

**Figura 1. Electrodo para electromiografía invasiva tipo aguja concéntrica desechable. Longitud de 3 - 4 cm de acero, con plástico o barniz aislante excepto en la punta.**



**Fuente:** Tomado de Medical Expo, Disponible en <http://www.medicalexpo.es/prod/ambu/product-69019-782906.html>

tología, por lo anterior se hace imposible utilizarlos en investigaciones para el desarrollo de prótesis.

**Electromiografía no invasiva:** También conocida como EMG superficial (sEMG), utiliza pequeños discos o conos generalmente de cloruro de plata (AgCl) con un gel conductor que ayuda a reducir la impedancia de la piel y además ayuda a adherir los electrodos. Debido a la comodidad de uso y simple ubicación, son la mejor opción en aplicaciones e investigaciones sobre prótesis mioeléctricas.

Una de sus desventajas, es que su eficacia depende de la posición en la que se coloca en el músculo, ya que detecta fácilmente las señales provenientes de los músculos superficiales y tiene dificultades para detectar señales de forma selectiva de músculos pequeños. Los electrodos de tipo de Ag/Ag-Cl con pasta de cloruro se utilizan de en este procedimiento, debido a sus propiedades de estabilidad y reducción del ruido. Con estos electrodos se puede obtener una idea de la electrogénesis global del músculo, pero no detectan potenciales de baja amplitud o de elevada frecuencia por lo cual su uso se

encuentra bastante restringido en EMG, se emplean para la determinación de latencias en la pruebas de conducción, fatiga muscular y en estudios kinesiológicos (Navarro, 2016).

En la membrana de las células donde se produce la diferencia de potencial, la corriente es transportada por iones, mientras que en el electrodo superficial, que es un dispositivo metálico altamente conductivo, la corriente se transporta por electrones. En consecuencia la conexión piel electrodo es muy ruidosa. Esto implica que la EMG de superficie sea más inexacta y necesite de complejos circuitos y filtros, tanto analógicos como digitales, para obtener la señal real proveniente del tejido muscular.

Uno de los puntos más críticos para el correcto funcionamiento de un sistema de identificación de señales electromiográficas, es el posicionamiento de los electrodos sobre el músculo. Es por eso que en 1996 surgió SENIAM (Surface Electromyography for non Invasive Assessment of Muscles) con el fin de establecer parámetros estándar como localización, tamaño y forma de los electrodos. Del resultado de trabajos europeos de SEMG se concluyó que las medidas adecuadas para el diámetro del electrodo son de 10 mm, que no existen mayores diferencias entre un electrodo circular y uno rectangular, siempre y cuando la superficie de ambos sea la misma, y ambos tipos tendrán valores similares de impedancia.

El SENIAM aconseja la preparación de la piel, que se hace con el fin de obtener un buen contacto con el electrodo y lograr menor interferencia eléctrica, menos riesgo de desequilibrio entre los electrodos y por lo tanto menor ruido. Entre las diferentes técnicas reportadas se encuentran: Afeitado, limpieza con alcohol, frotación con gel conductor y exfoliación de la piel con papel de lija (Hermens, 2000).

SENIAM proporciona recomendaciones para la selección del sensor y la ubicación del mismo. Los sensores deben colocarse en un lugar que proporcione estabilidad a la

señal. Los factores que pueden alterar la calidad de la señal en función de la posición son: *Presencia de puntos motores y/o tendones musculares y la presencia de otros músculos activos cerca del sensor SEMG (diafonía)*. Las recomendaciones para la colocación de los sensores son: con respecto a la *ubicación longitudinal* del sensor en el músculo, SENIAM recomienda colocar el sensor en la mitad de la zona (más) distal de la placa terminal del motor y el tendón distal. Con respecto a la *ubicación transversal* del sensor en el músculo, SENIAM recomienda colocar el sensor en la superficie lejos del 'borde' con otras subdivisiones o músculos para que la distancia geométrica del músculo a estas subdivisiones y otros músculos se maximice (Hermens, 2000).

Al colocar y fijar los electrodos, se debe tener en cuenta la distancia entre los ellos, el método de fijación y la ubicación del electrodo de referencia. *La distancia entre electrodos* se define como la distancia de centro a centro entre el área conductora de 2 electrodos bipolares. En general, se recomienda colocar los electrodos bipolares con una separación de 2 cm entre ellos. En músculos pequeños, la distancia entre los electrodos no debe exceder  $\frac{1}{4}$  de la longitud de la fibra muscular. Los electrodos de referencia deben colocarse en un tejido eléctrico inactivo. En la literatura se prefieren varias ubicaciones, dependiendo de los músculos desde los cuales se registra la Señal EMG. Las posiciones recomendadas son: *la muñeca, el codo o el tobillo*.

## **Procesamiento de Señales**

**Normalización.** Este procedimiento consiste en la modificación de la amplitud de las señales digitalizadas en un rango de entre 0 y 1, con el fin de evitar errores que pudieran ocurrir durante la toma de señales, o en el caso de estar generando una base de datos, evitar que la información que brindan las señales pueda verse afectada por *factores extrínsecos*, por ejemplo: posicionamiento de sensores, alteraciones en ganancias, cambios en equipo de toma de datos) o *factores intrínsecos* como: características

anatómicas y bioquímicas del sujeto de estudio (Bejarano, 2016).

**Segmentación.** Generalmente los sistemas de adquisición de señales toman continuamente todos los valores durante la creación de bases de datos, además de que no se puede asegurar por completo la repetitividad de los ejercicios de interés (Atzori, 2015). Estos protocolos traen como consecuencia que los datos adquiridos incluyen momentos de reposos de señales bioeléctricas, o patrones indeseados como leves movimientos de los pacientes. Por esta razón se hace necesario procedimientos que permitan obtener los segmentos deseados de estas señales.

Los métodos de segmentación pueden realizarse de manera *Manual* con la supervisión de un experto para realizar los respectivos recortes, o de forma *Automática*, con algoritmos basados en umbrales de activación o transformada Wavelet (Dominguez, 2015). El algoritmo basado en umbral aplica un filtro a la señal de tal manera que se eliminen las componentes de altas frecuencias pertenecientes a ruidos o cambios espontáneos en la señal que puedan activar el umbral. De esta manera cuando una unidad motora se activa y su actividad eléctrica sobrepasa el umbral, se acepta el segmento por encima del umbral como los valores de la señal de interés. El segundo algoritmo se basa en el análisis multiresolución aportado por la transformada Wavelet o transformada de Fourier con ventana, que permite observar la distribución de la energía en el dominio tiempo-frecuencia. Esta transformación permite distinguir los cambios a lo largo del tiempo e identificar los intervalos donde se concentran las frecuencias pertenecientes a la señal deseada.

**Caracterización.** En sistemas de reconocimiento de patrones de señales, se realiza esta etapa con el fin de obtener la información que brinda la señal tanto a lo largo del

tiempo como en su espectro de frecuencia. Del resultado de este procedimiento se obtienen unos parámetros  $x_{i,j}$  que describen particularidades que hacen posible identificar cada señal. Es así que se diferencian dos grupos de características:

- **Características temporales:** Hace referencia a las medidas que se pueden obtener de las señales en el dominio del tiempo  $f(t)$ , o cuantificadas y discretizadas cada  $T$  segundos (Proakis, 2007), de tal forma que de una señal se puede obtener su valor rms, valor medio, energía, área bajo la curva, longitud en el tiempo, desviación estándar y entropía.
- **Características espectrales:** La representación tiempo frecuencia de una señal provee información de la distribución de su energía en ambos dominios, obteniendo una descripción más completa del fenómeno fisiológico (Dominguez, 2015). Las técnicas más comunes usadas en la extracción del espectro de una señal son: *La transformada corta de Fourier (STFT)* y *la transformada Wavelet discreta y continua* (Romo, 2007). Por una parte, *la transformada de Fourier* es ampliamente utilizada en el procesamiento y análisis de señales. Los resultados obtenidos de la aplicación de este método son satisfactorios en los casos donde las señales son periódicas y suficientemente regulares. Los resultados varían cuando se analiza señales cuyo espectro varía con el tiempo (señales no estacionarias). Por otra parte, *la transformada Wavelet* es eficiente para el análisis local de señales no estacionarias de rápida transición debido a que provee un análisis multiresolución con una ventana dilatada.

Para obtener la transformada Wavelet ( $W_f$ ) de una función  $f(t)$ , se parte de una función wavelet llamada *wavelet madre*, que debe cumplir con las siguientes condiciones: energía finita y debe ser oscilatoria, es decir con valor medio igual a cero. Se obtienen las funciones wavelets a partir del corrimiento y cambio de escala de la función *wavelet madre*, como se define en la ecuación 1. Esta expresión define todas las versiones trasladadas y escaladas de la *wavelet madre*, y se les conoce como familia

wavelet.

$$\psi_{s,\tau}(t) = \frac{1}{\sqrt{s}}\psi\left(\frac{t-\tau}{s}\right), \quad (1)$$

donde  $\psi$  es la función *Wavelet madre*,  $s$  es el factor de escala y  $\tau$  es el factor de traslación (Dominguez, 2015).

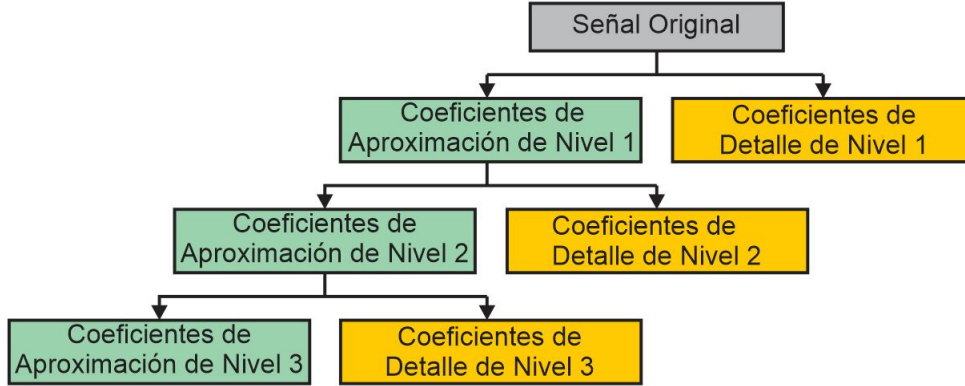
Se expresa la transformada wavelet ( $W_f(s, \tau)$ ) de una función  $f(t)$  continua, como el resultado del producto interno entre la propia señal y la familia wavelet. Tal como se expresa en la ecuación 2 (Sarría, 2007).

$$W_f(s, \tau) = \int f(t)\psi_{s,\tau}(t)dt \quad (2)$$

Los coeficientes Wavelet pueden ser calculados por un algoritmo discreto implementado de manera recursiva con filtros pasa altas y pasa bajas como el mostrado en la **Figura 2**, dando como resultado los coeficientes de aproximación y de detalle (Pinzón, 2009)(Canal, 2008).

**Selección de atributos** El proceso de caracterización tiene como fin adquirir la mayor cantidad de información posible, por lo que se llega a tener una gran cantidad de datos, y es necesario identificar aquellos subconjuntos de características que describen con mayor precisión las señales individuales. La selección de atributos permite agilizar los procesos de cómputo en la etapa de entrenamiento y evaluación, dando como resultado mejores tiempos de ejecución, evitando sobre aprendizaje y además clarifica cuál es la información repetitiva y cuál es la que permite diferenciar las señales EMG. Existen diferentes métodos de búsqueda que permiten realizar la selección de características, como son:

**Figura 2.** Descomposición Wavelet en 3 niveles. El cuadro gris representa la señal original, en el verde se representan los coeficientes de aproximación consecuentes del nivel anterior; finalmente en el cuadro amarillo los correspondientes coeficientes de detalle.



**Fuente:** Adaptado de (Canal, 2009).

- **Cfs Subset Eval:** Es un algoritmo de filtro simple que clasifica los subconjuntos de características según una función de evaluación heurística basada en correlación. El sesgo de la función de evaluación es hacia subconjuntos que contienen características que están altamente correlacionadas con la clase y no correlacionadas entre sí. Las características irrelevantes deben ignorarse porque tendrán una baja correlación con la clase. La aceptación de una característica dependerá de la medida en que predice las clases en áreas del espacio de instancia que aún no han sido predichas por otras características. La función de evaluación de este método es la mostrada en la ecuación 3.

$$M_S = \frac{r\overline{\tau_{cf}}}{\sqrt{r + r(r-1)\overline{\tau_{ff}}}}, \quad (3)$$

donde  $M_S$  es el mérito heurístico de un subconjunto  $S$  que contiene  $r$  características,  $\overline{\tau_{cf}}$  es el valor medio de las correlaciones entre clases y características, y  $\overline{\tau_{ff}}$  es el valor medio de las correlaciones entre características waikatonewzealand. Esta ecuación im-



pone una clasificación en subconjuntos de características en el espacio de búsqueda de todas las características posibles.

- **ReliefF:** Detecta las interacciones entre atributos para determinar los más relevantes y cuáles trabajan bien en grupos. Este proceso se realiza evaluando las características y aplicando técnicas de búsqueda heurísticas. El algoritmo iterativo consiste en seleccionar de manera aleatoria un dato de muestra, y luego selecciona la muestra más cercana de la misma clase *HIT*, y la más cerca de otra clase *MISS*. Se incrementa el peso de aquellos atributos que tienen valores similares para la instancia *HIT* y diferente para la instancia *MISS*, tal como se muestra a continuación (Urbanowicz, 2017).

$$P_i = P_i - (X_i - HIT)^2 + (X_i - MISS)^2, \quad (4)$$

donde  $X_i$  es un vector que contiene las características de un dato de muestra.

Se observa a partir de la ecuación 4, que el peso  $P_i$  de cualquier característica, aumenta si su valor es similar entre muestras cercanas de la misma clase; ocurre lo contrario si el valor de dicha característica difiere mucho. Tiene como desventaja que no puede detectar atributos redundantes (Lavalle, 2017).

- **Classifier Attribute Eval:** Evalúa el valor de un atributo, según si su pertenencia a un subconjunto de datos menor al conjunto original, contribuye a mejorar la clasificación con un clasificador específico.
- **Evaluación de atributos de ganancia de información:** Es un evaluador de atributos que calcula su valor midiendo la ganancia de información con respecto a la clase. Su propósito es descartar características irrelevantes. La información esperada  $I$  necesaria para clasificar una muestra determinada es dada por:.

$$I(S) = - \sum p_i \cdot \text{Log}_2(p_i), \quad (5)$$

donde  $p_i$  es la probabilidad para que una muestra arbitraria pertenezca a la clase  $y_i$  en un subconjunto de muestras  $S$ .

Se establece además la entropía, o información esperada ( $E$ ) basada en la partición en subconjuntos por el atributo  $A$ :

$$E(A) = - \sum I(S) \frac{s_{1,j} + s_{2,j} + \dots + s_{m,i}}{s}, \quad (6)$$

donde  $s_{i,j}$  es el número de muestras de la clase  $y_i$  en el subconjunto  $S_j$

La información que puede ser ganada  $G(A)$  por la bifurcación de  $A$  se define como:

$$G(A) = I(S) - E(A) \quad (7)$$

Finalmente se establece la relación de ganancia  $GR$  como:

$$GR = \text{Gain}(A) / \text{SplitInfo}(S), \quad (8)$$

donde  $\text{SplitInfo}(S)$  es la normalización de la ganancia de información (Karegowda, 2010).

**Aprendizaje de máquina** También conocido como *Machine Learning*, es un campo científico de la Inteligencia Artificial (IA), que brinda a las máquinas la capacidad de aprender a identificar patrones complejos en un conjunto de datos. El aprendizaje automático se enfoca en el desarrollo de algoritmos que sean capaces de optimizar su capacidad de realizar tareas específicas. Estos algoritmos tienen modelos definidos y su objetivo es optimizar los parámetros de dicho modelo a través de la experiencia (Alpaydin, 2010).

El aprendizaje automático puede abordarse de dos maneras diferentes, *Supervisada* y *No supervisada*. La primera consiste en tomar unos datos de muestra a los cuales un experto ha etiquetado (he ahí el porqué de su nombre “supervisado”), y la tarea del algoritmo es aprender el mapeo de la entrada a la salida, creado con el conjunto de entrenamiento y utilizarlos para predecir la clase de un nuevo dato de entrada. En el segundo tipo de aprendizaje, no se tiene un experto que le diga al algoritmo a que clase pertenece cada muestra, por lo tanto su tarea consiste en conocer la estructura de los datos de entrada con el fin de agruparlos, o estimar una distribución de probabilidad para encontrar regularidades entre ellos (Yannakakis, 2018).

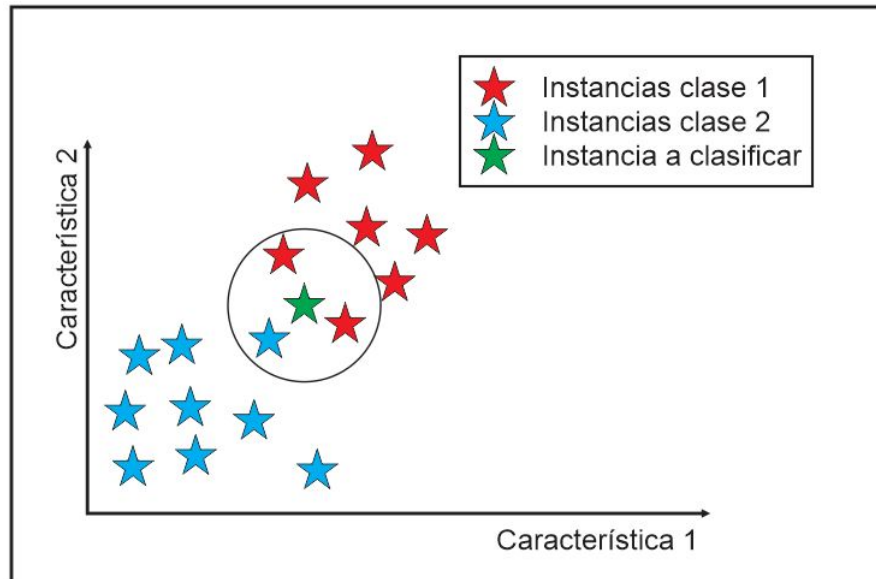
Entre los algoritmos más conocidos por su alto desempeño y capacidad de identificar múltiples clases se encuentran:

**Clasificador KNN:** El método de K vecinos más cercanos es un algoritmo no paramétrico de clasificación supervisada, basado en memoria y que no necesita un modelo para ajustarse. Este método asigna a un nuevo caso de entrada, el valor de la clase más frecuente a la que pertenecen sus  $k$  vecinos más cercanos (Alpaydin, 2010), como se observa en la figura **Figura 3**.

Existen diferentes configuraciones para este método de clasificación como se describen a continuación:

- **KNN con rechazo:** La idea básica es que para clasificar un nuevo caso se deben tener cuenta ciertas garantías, como por ejemplo que el número de votos a favor de una clase debe superar un umbral, también la asignación de un nuevo caso se hará cuando la diferencia de frecuencias mayor y segunda mayor sea superior a un valor predeterminado.
- **KNN con distancia media:** Asigna como valor de salida a un nuevo caso  $x$ , la

Figura 3. Modelo de clasificador KNN, donde la nueva instancia en color verde se clasifica como perteneciente a la clase 1, debido a que en sus 3 vecinos más cercanos la mayoría son muestras de la clase 1.



**Fuente:** Creación propia.

clase cuya distancia media de los  $k$  vecinos más cercanos sea menor. Se puede tener un problema biclase en el que la mayoría de las clases de los  $k$  vecinos más cercanos, sean de tipo “A”, pero se le asigna como salida “B” debido a que B tiene una menor distancia media con respecto a  $x$ .

- **KNN con pesos de variables:** En los anteriores modelos se calculaba la distancia entre el nuevo caso a clasificar con los datos de entrada con la distancia euclídea, así:

$$d(X_i, X_k) = \sqrt{\sum (X_i - X_k)^2}, \quad (9)$$

donde  $d(X_i, X_k)$  es la distancia eculidea entre los vectores  $X_i$  y  $X_k$ . Dichos vectores representan las muestras más cercanas y la muestra a clasificar respectivamente.

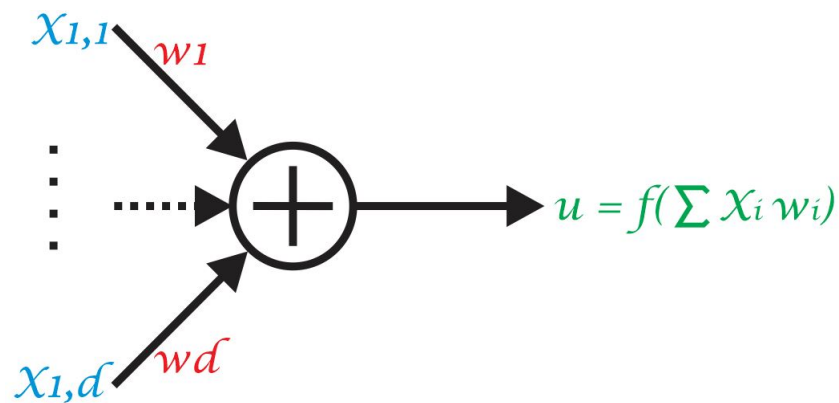
Esta manera de dar la distancia puede resultar perjudicial cuando algunas de las variables sean irrelevantes. Por esta razón se utiliza ponderaciones de la siguiente manera:

$$d(X_i, X_k) = \sqrt{\sum w_j (X_i - X_k)^2}. \quad (10)$$

dónde cada dato del conjunto original tiene un peso  $w_j$  que se calcula a partir de la medida de información mínima  $I(X_j, y)$  entre la componente  $X_j$  y la clase  $y$ , y se puede interpretar como la reducción en la incertidumbre sobre una de las variables cuando se conoce el valor de la otra.

**Redes neuronales artificiales:** Es una técnica de clasificación heurística que emula el comportamiento de un cerebro biológico, a través de una gran cantidad de neuronas artificiales que se conectan y activan mediante funciones (Alpaydin, 2010). Los modelos de redes neuronales de varias capas parten de un modelo básico conocido como perceptrón, y consta de una sola neurona como se representa en la **Figura 4**.

**Figura 4.** Modelo de perceptrón, donde la salida  $u$  se obtiene al aplicar la función sigmoideal a la sumatoria de las entradas  $x_{i,j}$  multiplicadas por los pesos  $w_i$ .



**Fuente:** Creación propia.

donde  $x_{i,j}$  denota los valores de entrada o características, y cada una de las  $n$  entradas tiene un peso asociado  $w$  (emulando el peso sináptico). Los valores de entrada son multiplicados por los respectivos pesos y sumados obteniendo:

$$u = w_1x_1 + w_2x_2 + \dots + w_nx_n = \sum_{i=1}^n w_i x_i. \quad (11)$$

En una red neuronal tipo perceptrón multicapa  $y$  es el valor de cada neurona y resulta de evaluar  $u$  en una función sigmoideal. La tarea de aprendizaje para este modelo, consiste en actualizar los pesos  $w$  con los datos de entrenamiento, de manera que se generen las salidas correctas.

En la etapa de entrenamiento de una red neuronal, generalmente se tiene el conjunto de entrenamiento completo, sin embargo los pesos se actualizan evaluando cada instancia por separado. Este enfoque resulta en el ahorro de costo de memoria, y el permitir actualizar los pesos debido a cambios en los datos de entrenamiento, por ejemplo en los cambios físicos que debe aprender a reconocer un detector de rostros, desgaste en sensores, un sistema de reconocimiento de voz, o los cambios físicos en las fibras musculares.

*Algoritmo Backpropagation:* Es un procedimiento para determinar los pesos  $w$  de manera que las salidas obtenidas coincidan con las deseadas, es decir se trata de minimizar el error:

$$E = \frac{1}{2} \sum_{k=1}^p \sum_{i=1}^M (z_i(k) - y_i(k))^2, \quad (12)$$

donde  $z_i$  es la salida obtenida por la red neuronal y  $y_i$  es la salida deseada.

El algoritmo de Backpropagation usa el método de descenso por el gradiente. Realiza el ajuste de pesos desde la capa de salida, según el error obtenido, y continúa con la propagación hacia las capas anteriores, hasta llegar a las unidades de entrada. Inicialmente se introduce los datos de entrada como estímulo para la capa de inicial, se determina los valores de las capas intermedias hasta llegar a las salidas estimadas, donde se compara

con las deseadas. A continuación se calcula el error para cada neurona de salida, y este valor es transmitido hacia atrás, por todas las capas intermedias y se van modificando los pesos según dicho error.

Suponiendo que se tiene la iteración  $k$ , donde según un dato de entrada cuya salida de unidad  $i$  es  $z_i(k)$ , y salida deseada  $y_i(k)$ , con pesos sinápticos  $w_{ij}(k)$  y  $t_{jr}(k)$ . Entonces la regla de modificación de los pesos  $w$  está dada por (Alpaydin, 2010):

$$w_{ij}(k+1) = w_{ij}(k) + \Delta w_{ij}(k), \quad (13)$$

donde  $\Delta w_{ij}(k)$  está dado por:

$$\Delta w_{ij}(k) = n \delta_i^2(k) s_j(k), \quad (14)$$

siendo  $n$ , un factor escalar llamado *Tasa de aprendizaje*,  $s_j(k)$  es el valor de la neurona y la función  $\delta_i^2$  se define por:

$$\delta_i^2 = g'(h_i) [z_i(k) - y_i(k)], \quad (15)$$

donde  $g'(h_i)$  es la derivada de la función de activación.

Para actualizar los pesos de la capa anterior, utilizando la regla de derivación en cadena, la variación de peso  $t_r$  está dada por:

$$\Delta t_{jr} = -n \frac{\partial E}{\partial t_{jr}(k)} = -n \frac{\partial E}{\partial s_j(k)} \frac{\partial s_j(k)}{\partial t_{jr}(k)} \quad (16)$$

sustituyendo las derivadas parciales la ecuación se convierte en:

$$= n \sum_{i=1}^M [z_i(k) - y_i(k)] g_1'(h_i) w_{ij}(k) g_2'(u_j) x_r(k), \quad (17)$$

reemplazando la ecuación 15 se obtiene:

$$n \sum_{i=1}^M \delta_i^2(k) w_{ij}(k) g_2'(u_j) x_r(k). \quad (18)$$

Se llama:

$$\delta_j^1(k) = g_2'(u_j) \sum_{i=1}^M w_{ij} \delta_i^2(k), \quad (19)$$

la variación de los pesos de la capa oculta vienen dados por:

$$\Delta t_{jr}(k) = n \delta_i^1(k) x_r(k). \quad (20)$$

**Parzen:** Es un clasificador basado en la estimación no paramétrica, que tiene por objetivo obtener estimaciones de densidad de probabilidad condicionada  $p(X_i/y)$ , sin tener conocimiento a priori de las mismas (Laencina, 2010). La estimación de probabilidad se realiza con el método de histograma, por eso se conoce al método parzen como una refinación del histograma.

Para realizar la estimación de probabilidad de un conjunto de muestras  $T$ s etiquetado, y uno sin etiquetar  $T$ . Se maximiza el logaritmo de verosimilitud del conjunto de entrenamiento  $T$ s variando el parámetro de suavizado  $\sigma_h$ , usando la estimación de *leave - one - out*. Es decir se busca maximizar la expresión:

$$\sum_{k=1}^Y \sum_{j=1}^{N_k} \ln(\hat{p}(X_{j,k}|y_k)), \quad (21)$$

donde  $X_{j,k}$  es la muestra  $j$  de la clase  $y_k$ ,  $Y$  es el conjunto de las clases esperadas  $y_k$  y  $N_k$  es el número de muestras con clase  $y_k$ . Y para cada muestra  $X$  en el conjunto de evaluación:

$$\hat{p}(X|y_k) = \frac{1}{N_k} \sum_{X_j \in T} \frac{1}{\sigma_h^N \sqrt{(2\pi)^N}} e^{\left(\frac{-\|X-X_j\|^2}{2\sigma_h^2}\right)}, \quad (22)$$

donde  $N$  es el número total de muestras. Además la asignación de la clase obtenida  $z = z_k$ . Donde  $k$  se elige como:

$$k = \operatorname{argmax}\{\hat{p}(X|y_i)\hat{p}(y_i)\} \quad (23)$$



# 1. METODOLOGÍA

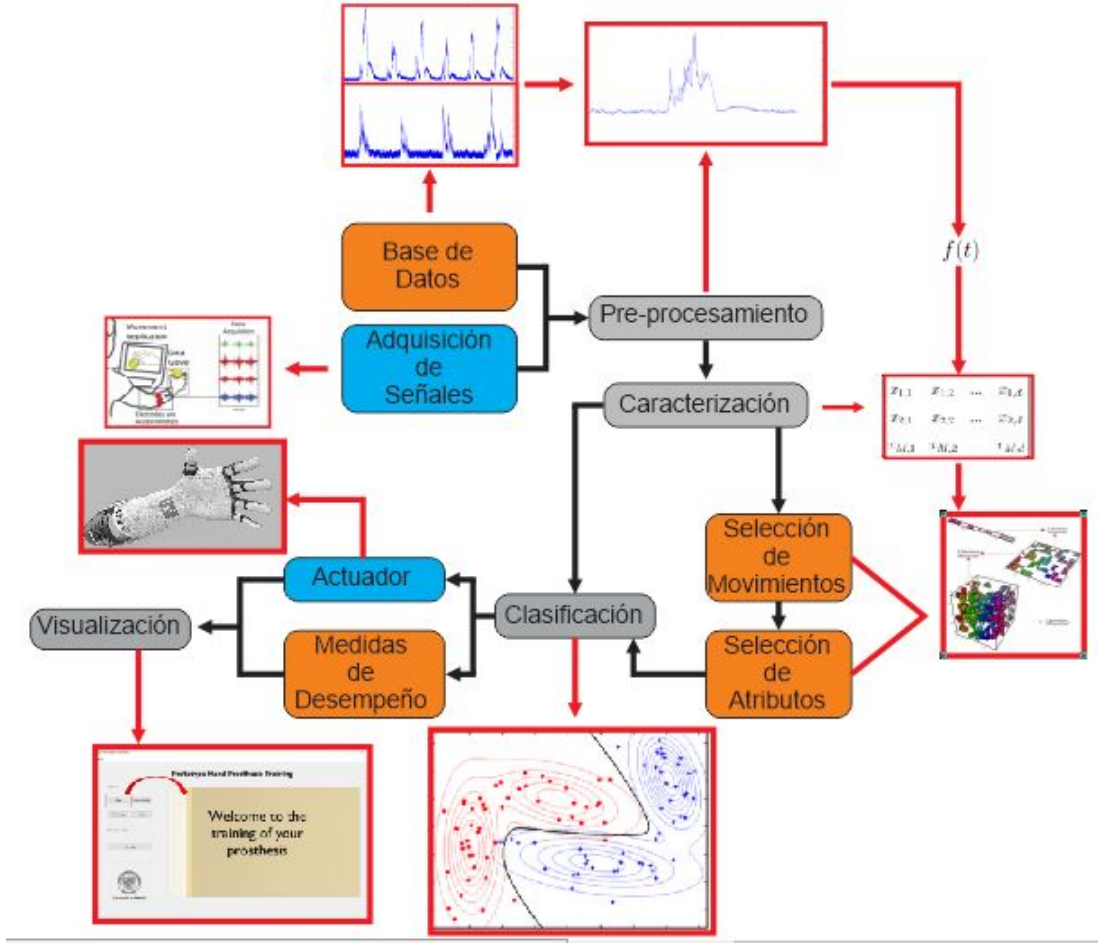
Para cumplir con los objetivos planteados en este proyecto de grado, se propuso la siguiente metodología que consta de las dos etapas: *Modelamiento* y *Aplicación*. El *Modelamiento* es el tratamiento de señales EMG con el fin de identificar movimientos mediante las mismas. En esta etapa se trabajó con una base de datos, se aplicaron pre-procesamiento, caracterización, se continuó con selección de movimientos, selección de características y finalmente medidas de desempeño. La *Aplicación* tuvo como fin la adquisición de señales EMG para realizar el control de un actuador, que en este caso fue un prototipo de prótesis de extremidad superior. El proceso comenzó con la adquisición de señales EMG, luego se pre-procesaron y caracterizaron, por último se entrenó un clasificador que toma la decisión sobre el movimiento que efectuó el actuador. Como se puede notar entre las dos etapas existieron procesos en común que son: pre procesamiento, caracterización y clasificación.

En la **Figura 5** se resumió la metodología. Los procesos que pertenecen a ambas se representaron con un color gris, los procesos que pertenecen a la etapa de *Modelamiento* tienen un color Naranja y los procesos de la etapa de *Aplicación* son de color azul.

## 1.1 Base de Datos

La base de datos considerada para la etapa de *Modelamiento* está disponible en el repositorio de NinaWeb, del proyecto *Ninapro* (Atzori, 2015). La base de datos mencionada, contiene la actividad electromiográfica de las extremidades superiores de 27 personas sanas, realizando 52 movimientos, divididos de la siguiente manera: 12 movimientos de dedos, 8 configuraciones isométricas e isotónicas de la mano, 9

Figura 5. Metodología para el estudio comparativo de técnicas *Machine Learning*. Los procesos que se compartieron entre la base de datos y toma de señales se rellenaron con color gris; los procesos de modelamiento se rellenaron con color naranja y los de aplicación en color azul.



**Fuente:** Original de esta investigación.

movimientos de muñeca y 23 movimientos de agarre funcionales. Los movimientos mencionados se muestran en la **Figura 6**.

Dichos movimientos fueron seleccionados de diferentes fuentes relevantes y guías de rehabilitación. La base de datos de *Ninapro* también incluye, un protocolo de adquisición

Figura 6. Movimientos base de datos Ninapro. Coloreados de azul los movimientos de dedos; en rosa 9 movimientos de muñeca y 8 configuraciones isométricas e isotónicas; y en verde 23 movimientos de agarre

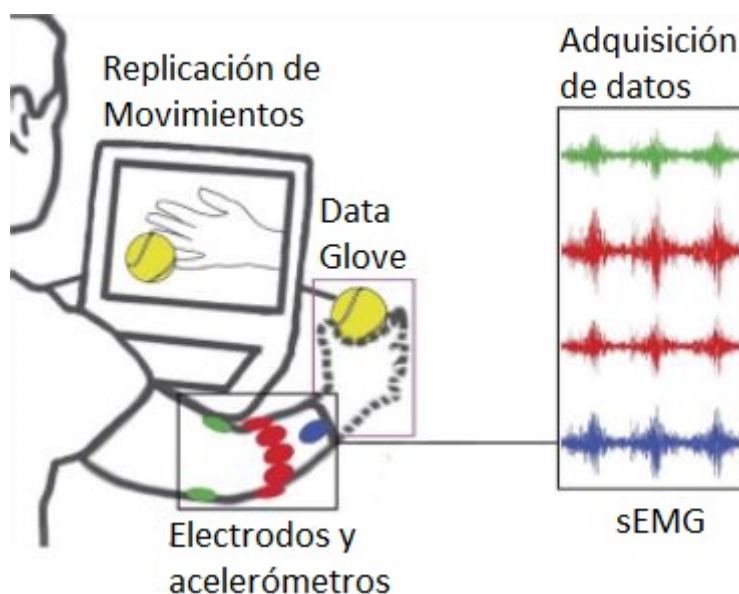
Ejercicio A			Ejercicio B			Ejercicio C						
1	Flexión de índice		13	Pulgar arriba		30	Agarre de diámetro grande		45	Agarre cuadrípode		
2	Extensión de índice		14	Extensión de medio e índice, flexión de los otros		31	Agarre de diámetro pequeño (agarre de fuerza)		46	Agarre lateral		
3	Flexión del corazón					32	Agarre de gancho fijo		47	Agarre de extensión paralela		
4	Extensión del corazón		15	Flexión de anular y meñique, extensión del resto		33	Agarre con extensión de dedo índice		48	Agarre tipo extensión		
5	Flexión del anular		16	Pulgar contrario base de meñique		34	Agarre medio		49	Agarre fuerte de disco		
6	Extensión del anular		17	Abducción de todos los dedos		35	Agarre de anillo		50	Abrir una botella con agarre tripode		
7	Flexión del meñique		18	Dedos flexionados juntos en puño		36	Agarre prismático de cuatro dedos		51	Gire un tornillo, destornillado con un agarre cilíndrico		
8	Extensión del meñique		19	Índice de puntero		37	Agarre cilíndrico		52	Cortar algo (agarre un cuchillo con un agarre de extensión del dedo índice)		
9	Aducción del pulgar		20	Aducción de los dedos extendidos		38	Agarre tripode de escritura					
10	Abducción del pulgar		21	Supinación de la muñeca (eje: dedo medio)		39	Agarre fuerte de esfera					
11	Flexión del pulgar		22	Pronación de la muñeca (eje: dedo medio)		40	Agarre de esfera con tres dedos					
12	Extensión del pulgar		23	Supinación de muñeca (eje: meñique)		41	Agarre de esfera de precisión					
			24	Pronación de muñeca (eje: dedo meñique)		42	Agarre de tripode					
			25	Flexión de la muñeca		43	Agarre prismático					
			26	Extensión de la muñeca		44	Agarre pequeño de punta					
			27	Desviación radial de la muñeca								
			28	Desviación cubital de muñeca								
			29	Extensión de muñeca con la mano cerrada								
											Descanso	

Fuente: Adaptación de Proyecto Ninapro de (Atzori, 2015).

y una breve descripción de los sujetos involucrados en la toma de datos. La actividad muscular del brazo se registró mediante 10 electrodos diferenciales. Se trabajó con una tasa de muestreo de 100 Hz. Los electrodos fueron equipados con un amplificador con un factor de 14000, un filtro pasabajos y un rectificador RMS. De los 10 electrodos, 8 fueron ubicados uniformemente alrededor del antebrazo, con una distancia constante, justo debajo del codo, y los dos electrodos restantes se colocan en el flexor y extensor largos del antebrazo (Atzori, 2014).

El protocolo experimental estableció que cada sujeto debe dar un consentimiento, llenar un breve cuestionario con información clínica, incluyendo edad, género, estatura, peso y un auto reporte del estado de salud. El voluntario se sentó frente una pantalla, donde se fueron mostrando los movimientos a repetir; mientras los electrodos, el DataGlove y el inclinómetro funcionan, el paciente repitió 10 veces cada uno de los 52 movimientos, como se puede observar en la **Figura 7**. Cada repetición dura 5 segundos, seguido de un periodo de descanso de 3 segundos.

**Figura 7.** Protocolo toma de señales base de datos. Se muestra a un sujeto con electrodos conectados en su antebrazo, siguiendo un vídeo en la pantalla y ejecutando movimientos mientras se registran las señales.



**Fuente:** Adaptado de (Atzori, 2014)

La base de datos, además de los registros electromiográficos, cuenta con un vector de etiquetas que indica que movimiento se realizó durante cada muestra de la señal. Con este vector se realizó el procedimiento de segmentación.

## 1.2 Adquisición de señales

Para la etapa de **Aplicación**, se llevó a cabo un procedimiento de adquisición de señales por medio 4 sensores musculares *Myoware* (AT-004-001), con 3 electrodos desechables como se muestra en la **Figura 8**. Este sensor está diseñado para ser conectado directamente a un microcontrolador, por lo tanto la señal de salida, no fue una señal pura EMG, sino una señal amplificada, rectificada e integrada. Se utilizaron 4 sensores ubicados en la mitad de los músculos *braquioradial*, *palmar largo*, *flexor superficial* y

*extensor común de los dedos* y los electrodos de referencia se ubicaron para todas las señales en la parte interna del antebrazo, a la altura de la muñeca, como se muestra en la figura **Figura 10**. Se utilizó una tasa de muestreo de 250Hz.

**Figura 8. Sensor muscular Myoware. Mide la actividad de un músculo al monitorear el potencial eléctrico generado por las células musculares.**



**Fuente:** MyoWare Muscle Sensor Kit [en línea]. Disponible en Internet: <https://www.sparkfun.com/products/13723>

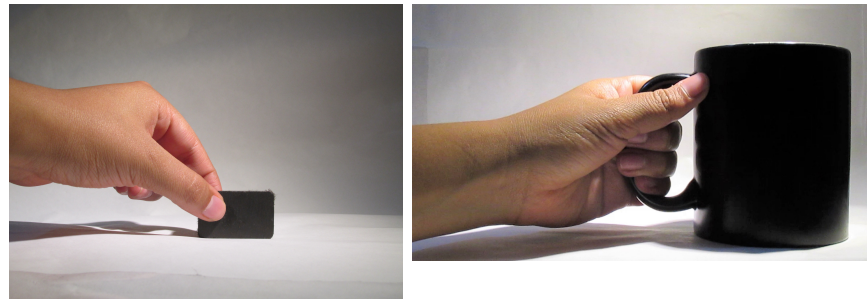
Se siguió un protocolo de limpieza de la piel, aplicación de gel conductor y posterior colocación de los sensores. El sujeto a quien se le tomó las señales permaneció sentado durante toda la prueba, aislado del piso, con la mano semiabierta como posición de descanso, el brazo y antebrazo formando un ángulo de 90 grados. Siguiendo los movimientos mostrados en un vídeo a través de una pantalla, realizaron quince repeticiones por cada movimiento, con un tiempo de descanso de 5 segundos por cada ejecución. En la **Figura 9** se muestra los movimientos ejecutados por cada usuario.

### 1.3 Pre-procesamiento

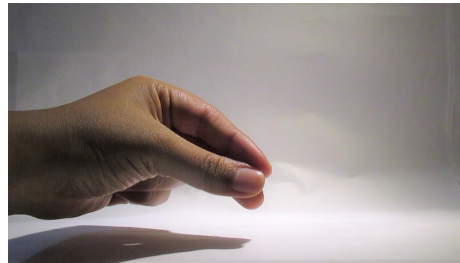
Por las razones descritas en la Sección de Marco Teórico - Procesamiento de Señales,

**Figura 9. Los 3 movimientos que ejecuta el usuario, a) Movimiento 1: Agarre pequeño, b) Movimiento 2: Agarre lateral y c) Movimiento 3: Agarre de tripode de escritura.**

(a) Movimiento 1: Agarre pequeño. (b) Movimiento 2: Agarre lateral.



(c) Movimiento 3: Agarre de tripode de escritura.



**Fuente:** Propia de esta investigación.

se realizó un proceso de normalización por cada canal presente en la base de datos, acotando la señal entre valores de 0 y 1. Un procedimiento idéntico se realizó con las señales tomadas con los sensores MyoWare. Adicionalmente se procedió a segmentar las señales con el vector de etiquetas. Las señales electromiográficas recortadas y normalizadas se almacenaron en una estructura de datos.

## 1.4 Caracterización

Al realizar el pre-procesamiento a las señales EMG, fue posible tratarlas como un con-

**Figura 10. Posicionamiento de los electrodos alrededor del antebrazo para toma de señales y posterior entrenamiento de clasificadores.**

(a) Vista superior del brazo con los electrodos ubicados. (b) Vista lateral izquierda de brazo del voluntario.



(c) Vista lateral derecha del voluntario, con electrodos ubicados. (d) Vista posterior del brazo del voluntario con electrodos de referencia en la muñeca



**Fuente:** Original de esta investigación.

junto de datos que pueden ser representados por una serie de características. Con base en trabajos e investigaciones anteriores, se extrajeron dos tipos de características: las *Temporales* y *Espectrales* que se extrajeron a cada segmento de señal, de cada repetición  $F[n]$ .



Las características temporales tomadas en cuenta fueron: Área bajo la curva, valor medio absoluto, valor RMS, desviación estándar, varianza, mediana, entropía, coeficiente de curtosis, coeficiente de fischer, amplitud de Willison (WAMP), histograma y longitud de la forma de onda (Phinyomark, 2009)(Han, 2000) (Negi,2016)(Romo, 2007)(Phinyomark, 2011)

Para la extracción de características espectrales, dos métodos fueron utilizados: *Transformada de Fourier* y *Transformada Wavelet* (Ahlstrom, 2006)(Krishna, 2015) (Phinyomark, 2009)(Romo, 2007)(Podrug,2015). Con la transformada de Fourier se obtuvo los valores de frecuencia pico y frecuencia media. La descomposición espectral de las señales se realizó por medio de la transformada wavelet discreta disponible en el software Matlab. Se utilizó Daubechies2 como función Wavelet madre, con tres niveles de descomposición. A los coeficientes obtenidos con la descomposición Wavelet se les obtuvo medidas descritas en las características temporales. Siendo  $F[n]$  los segmentos de señales EMG, las ecuaciones de las características mencionadas se consignaron en la **Tabla 1**.

Finalmente al aplicar las medidas de características temporales a los coeficientes Wavelets se obtuvieron 45 características diferentes para las señales EMG. Estas 45 características se extrajeron de cada uno de los 10 canales de la base de datos. Los valores obtenidos se organizaron en una matriz, definida como:  $X \in R^{M \times d}$ , donde  $M$  es el número de muestras igual a 14040, correspondiente a las 270 repeticiones ( $k$ ) por las 52 clases ( $y$ ).El valor de  $d = 450$  corresponde al total de características. Esta matriz se muestra de una manera más clara en la matriz de la ecuación 24.

$$\mathbf{X} = [x_{i,j}] = \begin{bmatrix} x_{1,1} & x_{1,2} & \dots & x_{1,d} \\ x_{2,1} & x_{2,2} & \dots & x_{2,d} \\ x_{M,1} & x_{M,2} & \dots & x_{M,d} \end{bmatrix} \quad (24)$$

**Tabla 1. Formulas utilizadas para caracterización**

Característica	Fórmula
Área bajo la curva	$\sum_{n=1}^N F[n]$
Valor Medio Absoluto	$\frac{1}{N} \sum  F[n] $
Valor RMS	$\sqrt{\frac{1}{N} \sum F[n]^2}$
Varianza	$\frac{1}{N-1} \sum (F[n] - \bar{F})^2$
Desviación estándar	$\sqrt{\frac{1}{N-1} \sum (F[n] - \bar{F})^2}$
Entropía	$\sum P \cdot \log_2(P)$
Mediana	$F_{\frac{n+1}{2}}, \text{ para } n \text{ impar}$
	$\frac{F_{\frac{n}{2}} + F_{\frac{n}{2}+1}}{2}, \text{ para } n \text{ par}$
Coeficiente de Curtosis	$\frac{\sum (F[n] - \bar{F})^4}{Ns^4} - 3$
Coeficiente de asimetría de Fischer	$\frac{\sum (F[n] - \bar{F})^3}{Ns^3}$
Amplitud de Willison	$\sum f(F[n] - F[n+1])$ $f(F[n]) = \begin{cases} 1 & \text{si } F[n] \geq \text{umbral} \\ 0 & \text{si } F[n] < \text{umbral} \end{cases}$
Longitud de la forma de onda	$\sum  (F[n+1] - F[n]) $

## 1.5 Selección de movimientos

Como ya se ha mencionado anteriormente, la base de datos cuenta con 52 movimientos, cada uno de ellos se pudieron representar por un grupo de valores extraídos de las señales EMG. Existieron movimientos que tienen características muy similares, y esto perjudicó el desempeño de los clasificadores. Por lo anterior, fue necesario seleccionar los movimientos que sean más diferenciables entre sí. Para esto se propuso un criterio de selección basado en la distancia euclidiana ( $DE$ ) y la desviación estándar ( $SD$ ). Tomando la matriz de características, se calculó el centro de la nube de puntos ( $NP$ ) para cada movimiento, como el valor medio de las repeticiones  $k$ , pertenecientes a cada

clase.

La distancia euclidiana entre dos clases se calculó como la raíz de las diferencias cuadradas entre las coordenadas de un par de centros. La distancia euclidiana entre los centroides permitió evaluar qué tan separadas están las clases, y la desviación estándar, dijo que tan compacta era una nube de puntos. Para mejorar el rendimiento de la clasificación, fue necesario que las  $NP$  de cada movimiento estuvieran lo más separadas y compactas posible. Por lo tanto, un parámetro  $Ms$  que permitió evaluar estas dos características entre dos clases diferentes se definió como:

$$Ms(y_1, y_2) = \frac{DE(y_1, y_2)}{SD_1 + SD_2} \quad (25)$$

Donde  $DE(y_1, y_2)$  es la distancia euclidiana entre los centroides de la clase 1 ( $y_1$ ) y la clase 2 ( $y_2$ ),  $SD_1$  y  $SD_2$  son la desviación estándar de las clases 1 y 2. La  $SD$  debe ser inversamente proporcional al parámetro  $Ms$  para maximizar y mejorar el proceso de clasificación. Esta ecuación se aplicó a todas las combinaciones de clases. Los movimientos que se seleccionó son aquellos donde el parámetro  $Ms$  sea mayor. Los 12 movimientos más separables obtenidos con este procedimiento fueron: 4, 5, 6, 19, 25, 26, 27, 38, 44, 46, 49 y 51, se presentan en **Figura 11**.

## 1.6 Selección de características

A pesar de disminuir el número de movimientos para trabajar, se manejaba una alta cantidad de variables. Mucha información extraída de las señales EMG es redundante o inútil, y se hizo necesario disminuir el número de variables para así mejorar el rendimiento de los clasificadores.

Para llevar a cabo esta etapa, se utilizó el software Weka, que cuenta con los algoritmos para aplicar diferentes técnicas de selección de atributos. Se aplicó Cfs Subset Eval,

**Figura 11. Movimientos de la base de datos que presentaron mayor separabilidad.**



**Fuente:** Creación propia.

Classifier Attribute Eval, Correlation Attribute Eval, Gain Ratio Attribute Eval y el algoritmo ReliefF.

Estos algoritmos organizaron las características, pero no dicen cuántas eran suficientes para llevar a cabo un procedimiento de clasificación óptimo. Para decidir qué cantidad de atributos eran apropiados, se realizó la siguiente prueba: Con los clasificadores KNN, Parzen y Red Neuronal con algoritmo de Backpropagation, se tomó un conjunto inicial de características, se calculó el error de clasificación promedio de 10 iteraciones, y se aumentó el número de características de acuerdo al orden mostrado por el algoritmo de selección de atributos.

## 1.7 Clasificación

Para la base de datos, el proceso de clasificación de los movimientos se llevó a cabo por medio de los algoritmos de KNN simple, ANN entrenada con algoritmo de retropropagación y el clasificador probabilístico de parzen, descritos en la Sección de Marco Teórico - Aprendizaje de máquina. Para las señales adquiridas, las cuales fueron tomadas después de hacer los estudios y pruebas con la base de datos, se decidió realizar la clasificación con los algoritmos de KNN simple y ANN entrenada con algoritmo de retropropagación, ya que su desempeño con la base de datos fue muy alta. Estos resultados se explicaron de manera más profunda en la Sección de Resultados.

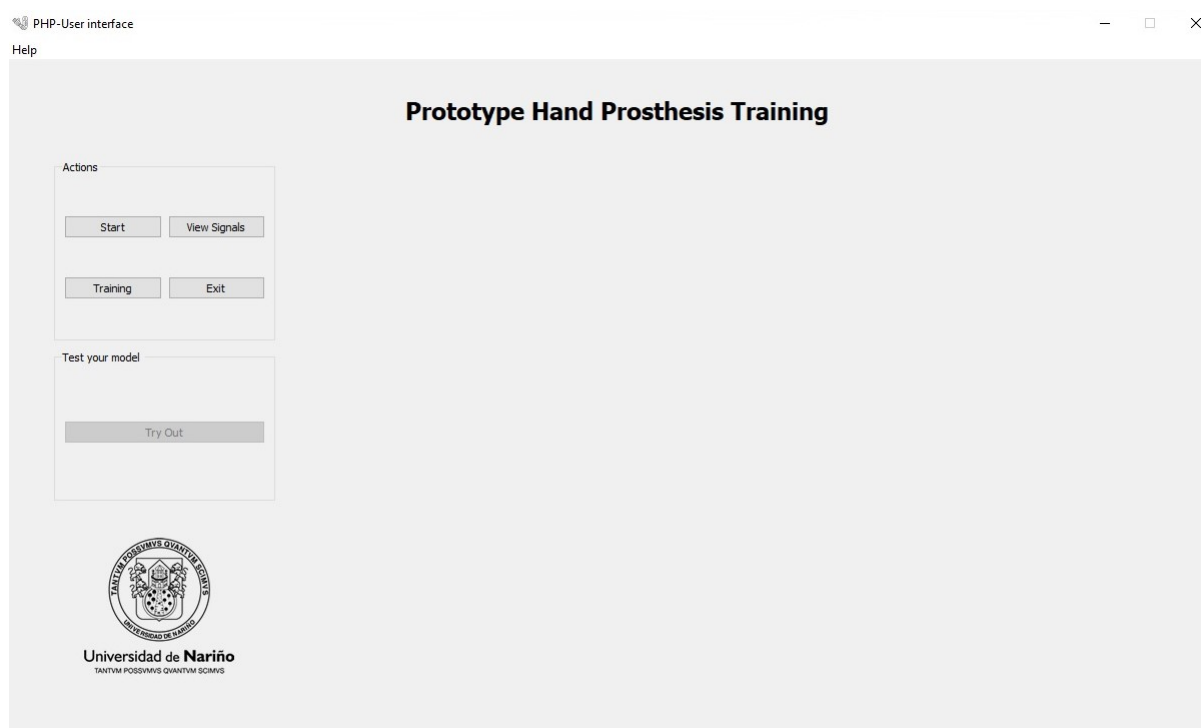
## 1.8 Visualización

Para complementar este proyecto, se realizó una interfaz interactiva. En esta interfaz el usuario entrenó un modelo ya sea KNN o ANN, con sus señales EMG, para luego controlar el prototipo de prótesis. Esta interfaz fue realizada en python con la ayuda de PyQt Designer.

Como se puede observar en la **Figura 12**, la ventana principal de la interfaz, tiene 5 botones: *Start*, *View Signals*, *Training*, *Exit* y *Try Out*. El uso de la interfaz se centró en 3 etapas: *Inicio*, *Entrenamiento* y *Prueba de los modelos* que se explican con mayor profundidad en los siguientes párrafos.

**1.8.1 Inicio.** Vinculado con el botón *Start*, al ser presionado comenzó con dos procesos en paralelo. Por un lado se reproduce un vídeo que mostraba al usuario los movimientos que debía que repetir y el momento en que debía hacerlo. Mientras se realizaban las repeticiones, se tomaban las señales de los sensores MyoWare previamente ubicados y se guardaron en un txt con la fecha y hora de inicio para su uso en la etapa

**Figura 12.** Vista pantalla principal de la interfaz para el entrenamiento de clasificadores.



**Fuente:** Original de esta investigación.

de entrenamiento.

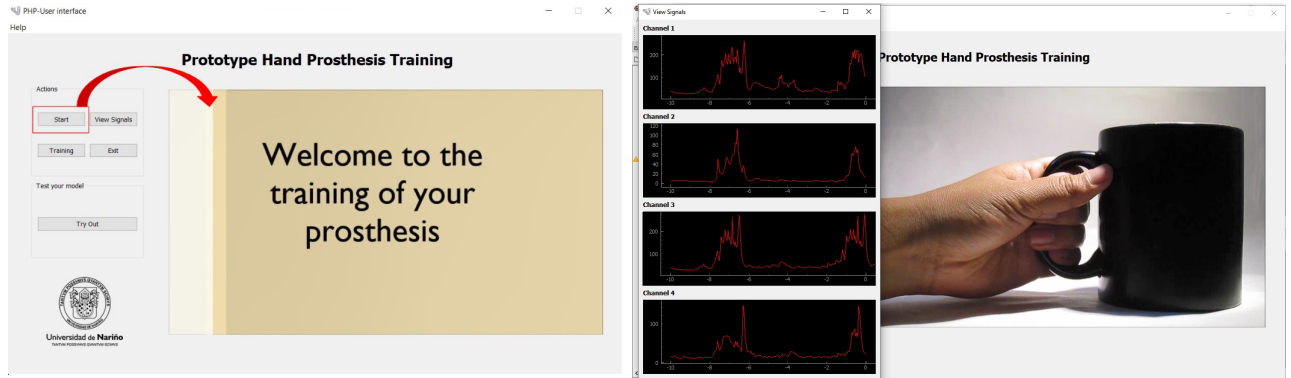
La toma de los datos se realizó a través de un arduino uno, mediante comunicación serial a una velocidad de transmisión de 115200 baudios y con 1000 Hz como frecuencia de muestreo. Por otro lado, al presionar *View Signals* se podía observar en tiempo real las señales EMG que se producían en los 4 sensores como indica la **Figura 13**.

**1.8.2 Entrenamiento.** Al finalizar el video, con el boton *Training*, se pudo seguir con la siguiente etapa. En una nueva ventana se presentaba la señal adquirida previa-

**Figura 13. Etapa de inicio de la interfaz gráfica.**

(a) Reproducción del video que muestra los movimientos a repetir.

(b) Visualización de señales captadas por los sensores MyoWare ejecutando un movimiento



**Fuente:** Original de esta investigación.

mente como se muestra en **Figura 14a**, donde el usuario debía seleccionar las señales que pertenecían a los 3 movimientos, **Figura 14b**.

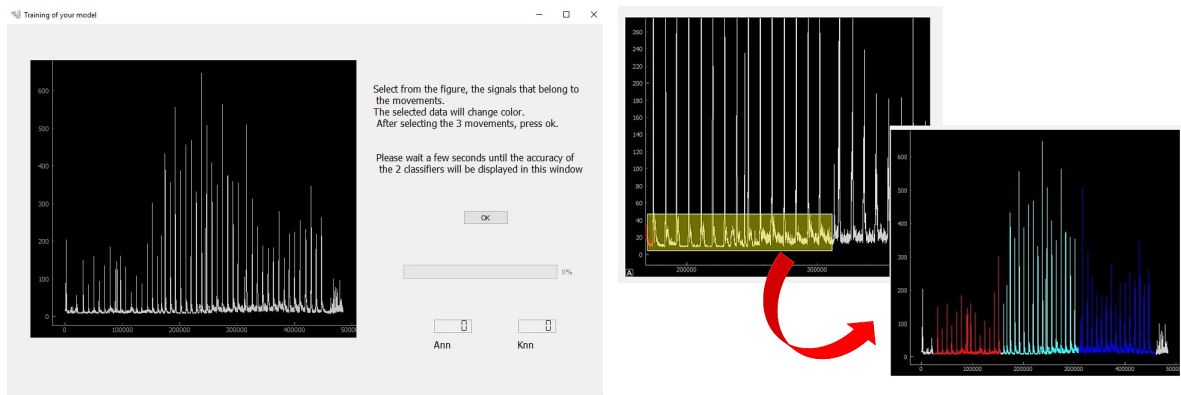
Se segmentó las señales y se procedió a caracterizar, luego se realizó el entrenamiento con KNN y ANN. Los parámetros de estos clasificadores se optimizaron hasta obtener un rendimiento adecuado. Finalmente se mostró al usuario la exactitud de los clasificadores y una gráfica de las 3 mejores características según ReliefF, representando los 3 movimientos **Figura 14d**.

**1.8.3 Prueba de los modelos.** Al presionar el botón *Try Out*, se abrió una ventana auxiliar donde se tomó una nueva señal EMG, ejecutando uno de los 3 movimientos. Se realizó el mismo proceso que a las primeras señales y se predijo el movimiento con el modelo entrenado. La orden de ejecución de control se envió vía bluetooth, para que el actuador ejecutara dicho movimiento.

**Figura 14. Etapa de Entrenamiento en la interfaz gráfica. Se recortaron las señales EMG adquiridas, se caracterizaron y se entrenaron los clasificadores KNN Y ANN.**

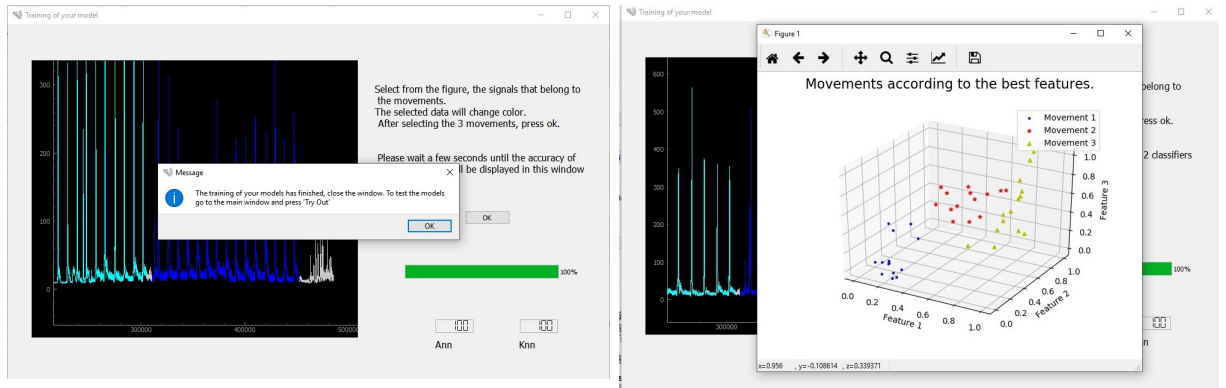
(a) Ventana de Training, mostrando la señal

EMG adquirida mientras el usuario ejecutaba los (b) Selección de las señales que pertenecían a los 3 movimientos.



(c) Entrenamiento de los clasificadores finalizado, se

(d) Finalización de la etapa Training, mostrando la gráfica de mejores características según Relief



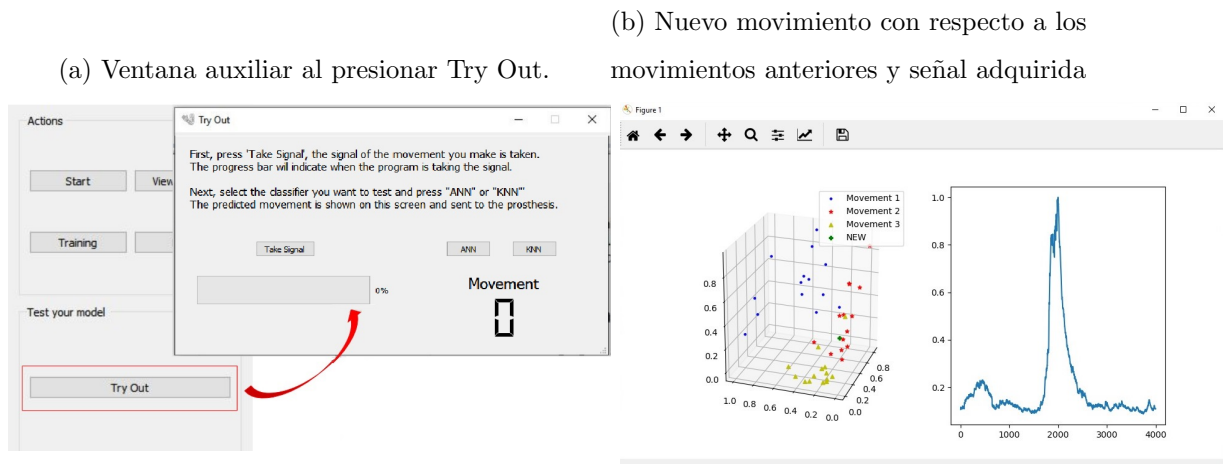
**Fuente:** Original de esta investigación.

Finalmente se presentó una gráfica donde se mostraba la nube de puntos de los 3 movimientos de la etapa de entrenamiento, la instancia de la nueva señal en un plot



tipo 3D y la señal tomada. Esta etapa se muestra en la **Figura 15**.

**Figura 15.** Etapa de prueba de modelo en la interfaz gráfica, tomando una nueva señal y realizando predicción con modelos entrenados.



**Fuente:** Original de esta investigación.

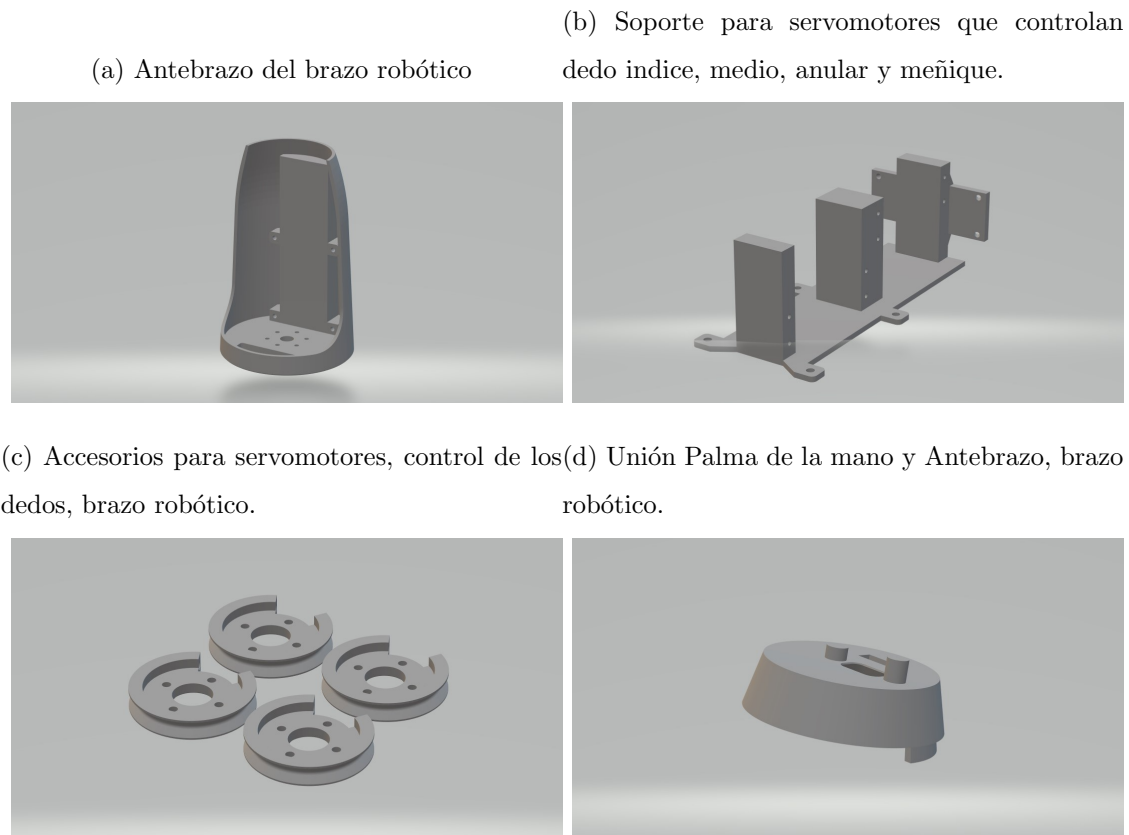
## 1.9 Actuador

Se implementó un diseño de prototipo de prótesis de extremidad superior, que cumplía con los requisitos de movilidad básica y grados de libertad que este proyecto necesitaba: movimiento independiente de los 5 dedos controlados mediante servo motores y giro del brazo, igualmente controlado por un servomotor. El diseño constó de 24 piezas, incluyendo, el antebrazo y la palma de la mano, entre otros.

La primera parte es el antebrazo, mostrada en la **Figura 16a**, que fue la pieza más grande del diseño, permitiendo que se acoplen el motor de giro y los servomotores que controlan los dedos. En **Figura 16b**, se encuentra la estructura que sostuvo los 4 servomotores, encargados del movimiento de los dedos índice, medio, anular y meñique.

En **Figura 16c** se mostraron acoples para los servomotores, que permitieron insertar nylon para contraer y estirar los dedos del brazo robótico.

**Figura 16. Modelo 3D de las partes del brazo robótico, a) Antebrazo, b) Soporte para servomotores, c) Accesorio para servomotores y d) Unión Palma de la mano y Antebrazo.**

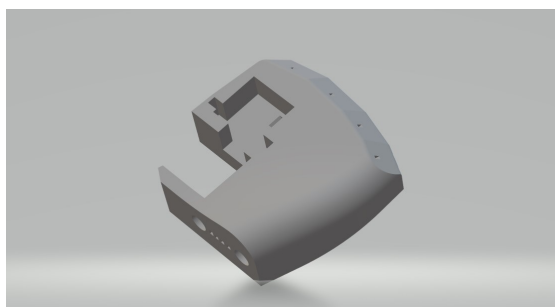


**Fuente:** GROSS, Ryan. Humanoid Robotic Hand [en línea]. Disponible en <https://www.myminifactory.com/es/object/3d-print-humanoid-robotic-hand-34508>.

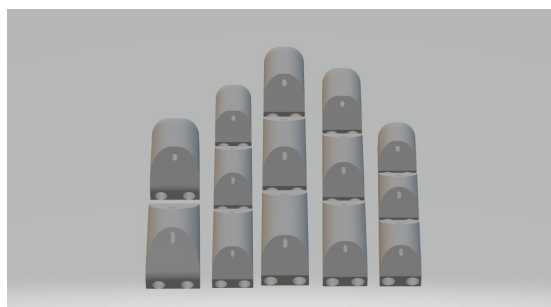
Para unir la palma con el antebrazo, se utilizó la pieza de la **Figura 16d**, la cual se aseguró con la parte superior del antebrazo. Como se puede observar en la **Figura 17a**, en la parte superior de la palma se acoplaron los dedos índice, medio, anular y meñique, y en la parte izquierda el dedo pulgar con su servo motor.

**Figura 17.** Modelo 3D de las partes del brazo robótico implementado, a)Palma de la mano del brazo robótico, b)Dedos índice, medio, anular y meñique, c )Acople dedo pulgar con servomotor y d) Acople servomotor con Antebrazo para giro.

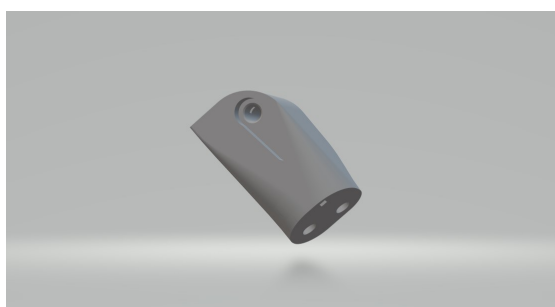
(a) Palma del brazo robótico



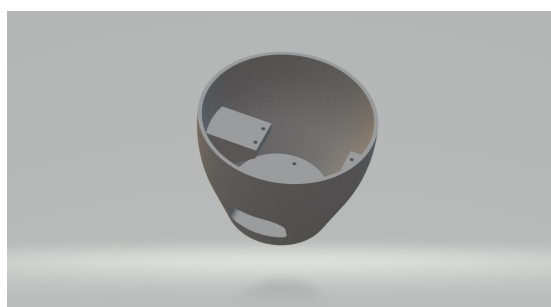
(b) Dedos índice, medio, anular y meñique, e brazo robotico.



(c) Acople dedo pulgar a servomotor del brazo robótico.



(d) Acople servomotor con Antebrazo para giro.



**Fuente:** GROSS, Ryan. Humanoid Robotic Hand [en línea]. Disponible en <https://www.myminifactory.com/es/object/3d-print-humanoid-robotic-hand-34508>.

En la **Figura 17b** y **Figura 17c**, se puede apreciar los diseños en 3D, el acople de dedo pulgar y las falanges de todos los dedos. Finalmente, la base para el motor permitió el giro de todo el brazo, se acoplaron en la parte inferior del antebrazo. Su modelo 3D se consignó en la **Figura 17d**. Finalmente, se observa en **Figura 18** el brazo robótico que se utilizó en este proyecto, fue impreso con los modelos 3D ya mencionados.

Al trabajar con servomotores, el control de movimiento se realizó a través de la tarjeta

**Figura 18. Brazo robótico impreso en 3D y completamente armado.**



**Fuente:** Creación propia.

ESP32, la cual tiene un módulo bluetooth que permitió la comunicación con la interfaz gráfica, recibiendo qué movimiento debía ejecutar según la predicción realizada . En esta tarjeta se ejecutó un programa en el cual se definió cada uno de los 3 movimientos y un descanso. Al iniciar el programa siempre se puso a los motores en posición de descanso, si se recibía un movimiento, este se ejecutaba durante 5 segundos y luego vuelve a la posición de descanso. Para la alimentación de los servomotores se utilizó una batería de litio de 11.1 voltios, se realizó una etapa de regulación de voltaje mediante la tarjeta LM2596 y para la tarjeta un cargador de 5 voltios.

## 2. MARCO EXPERIMENTAL

En esta sección se describen los experimentos llevados a cabo para lograr identificar un conjunto de características relevantes de las señales EMG, y una posterior comparación de los clasificadores KNN, ANN y Parzen.

### 2.1 Medidas de desempeño

La evaluación de la efectividad de clasificación se hizo por medio de las medidas de desempeño de sensibilidad ( $Se$ ), especificidad ( $Es$ ) y porcentaje de clasificación ( $Pc$ ), definidas como se muestra a continuación:

$$Se = \frac{Vp}{Vp + Fn}. \quad (26)$$

$$Sp = \frac{Vn}{Vn + Fp}. \quad (27)$$

$$Pc = \frac{Vn + Vp}{Vp + Vn + Fp + Fn} \quad (28)$$

donde:

$Vp$ : Son llamados verdaderos positivos, o casos de la clase de interés clasificados correctamente.

$Vn$ : Son los verdaderos negativos, o casos diferentes a la clase de interés clasificados correctamente.

$Fp$ : Falsos positivos o casos diferentes a la clase de interés clasificados como la clase de interés.

$Fn$ : Falsos negativos o casos de la clase de interés clasificados como casos diferentes de la clase de interés.

**Matriz de confusión** Una matriz de confusión es un método de evaluación de un modelo de clasificación supervisado. Permite la visualización de la cantidad de muestras clasificadas en cada clase. Esta información se organizó en una matriz cuadrada de  $n \times n$ , donde  $n$  es el número de clases. Las columnas de esta matriz corresponden a las predicciones realizadas por el modelo, mientras que las filas corresponden a la clase real. En la **Figura 19** se muestra la matriz de confusión para una clasificación multiclase.

**Figura 19.** Matriz de confusión.

		Predicción del Modelo			
		Clase 1	Clase 2	...	Clase n
Clase real	Clase 1	Vp1	E12	...	E1n
	Clase 2	E21	Vp2	...	E2n
	⋮	⋮	⋮	⋮	⋮
	Clase n	En1	En2	...	Vpn

**Fuente:** Creación propia.

Los valores de la diagonal con casillas de color verde son aquellos bien clasificados, y donde  $E_{ij}$  son los elementos de la clase  $i$  clasificados como elementos de de la clase  $j$ .

## 2.2 Pruebas aplicadas a la base de datos.

En la Sección de Metodología se explicó los procedimientos realizados con la base de datos y las señales adquiridas. En esta sección se describen los experimentos llevados a cabo para la identificar las características descriptivas y el clasificador con mejor desempeño.

Las pruebas de selección de atributos se realizaron en el software Weka. Las pruebas de clasificación se realizaron en el software Matlab, con las herramientas de Prtools. Se dividió entonces la sección de experimentación con la base de datos en dos secciones:

**Selección de atributos:** como se explicó en la sección de Metodología, para esta sección se utilizaron 5 métodos de selección de atributos. Cada método arrojó un vector con las características ordenadas según su relevancia. Se definieron entonces los siguientes experimentos con el fin de determinar las características relevantes y concentración de la información:

- **Experimento 1:** Se seleccionó las mejores características según el algoritmo CF-SSUBSETEVAL del software Weka. Se mostró como resultado en la **Figura 20** el histograma de frecuencias de los 82 atributos arrojados.
- **Experimento 2:** Se seleccionó las mejores características según el algoritmo CLASSIFIER ATTRIBUTE EVAL del software Weka. Se mostró como resultado en la **Figura 21** el histograma de frecuencias de los 100 mejores atributos arrojados.
- **Experimento 3:** Se seleccionó las mejores características según el algoritmo CORRELATION ATTRIBUTE EVAL del software Weka. Se mostró como resultado en la **Figura 22** el histograma de frecuencias de los 100 mejores atributos arrojados.
- **Experimento 4:** Se seleccionó las mejores características según el algoritmo GAIN RATIO ATTRIBUTE EVAL del software Weka. Se mostró como resultado en la **Figura 23** el histograma de frecuencias de los 100 mejores atributos arrojados.
- **Experimento 5:** Se seleccionó las mejores características según el algoritmo RELIEF del software Weka. Se mostró como resultado en la **Figura 24** el histograma de frecuencias de los 100 mejores atributos arrojados.

**Comparación de clasificadores:** En cada experimento se calculó el promedio de error de clasificación variando el número de características. El siguiente procedimiento

se realizó para descartar que las evaluaciones de desempeño fueran el resultado de errores en las muestras: Seleccionando de manera aleatoria conjuntos balanceados de cada clase, se evaluó cada clasificador y se repitió el procedimiento diez veces; al finalizar se guardó las medidas de desempeño. Además se obtuvo las medidas de sensibilidad promedio y especificidad promedio del conjunto de características con mejor desempeño. También se almacenó la matriz de confusión del modelo con mejor porcentaje de clasificación. Se definió entonces los siguientes experimentos:

- **Experimento 6:** Clasificación con KNN variando el número de características según el orden arrojado por Cfs Subset Eval.
- **Experimento 7:** Clasificación con KNN variando el número de características según el orden arrojado por Classifier Attribute Eval.
- **Experimento 8:** Clasificación con KNN variando el número de características según el orden arrojado por Correlation Attribute Eval.
- **Experimento 9:** Clasificación con KNN variando el número de características según el orden arrojado por Gain Ratio Attribute Eval.
- **Experimento 10:** Clasificación con KNN variando el número de características según el orden arrojado por Relief.
- **Experimento 11:** Clasificación con Parzen variando el número de características según el orden arrojado por Cfs Subset Eval.
- **Experimento 12:** Clasificación con Parzen variando el número de características según el orden arrojado por Classifier Attribute Eval.
- **Experimento 13:** Clasificación con Parzen variando el número de características según el orden arrojado por Correlation Attribute Eval.
- **Experimento 14:** Clasificación con Parzen variando el número de características según el orden arrojado por Gain Ratio Attribute Eval.



- **Experimento 15:** Clasificación con Parzen variando el número de características según el orden arrojado por Relief.
- **Experimento 16:** Clasificación con ANN variando el número de características según el orden arrojado por Cfs Subset Eval.
- **Experimento 17:** Clasificación con ANN variando el número de características según el orden arrojado por Classifier Attribute Eval.
- **Experimento 18:** Clasificación con ANN variando el número de características según el orden arrojado por Correlation Attribute Eval.
- **Experimento 19:** Clasificación con ANN variando el número de características según el orden arrojado por Gain Ratio Attribute Eval.
- **Experimento 20:** Clasificación con ANN variando el número de características según el orden arrojado por Relief.

## 2.3 Prueba de Interfaz gráfica y Actuador

Se planteó probar tanto la interfaz gráfica, cada una de sus etapas descritas en la sección de Metodología - Visualización, y como el actuador respondió a las órdenes de la interfaz y como ejecutó cada uno de los 3 movimientos. Un voluntario efectuó cada una de las etapas de la interfaz gráfica: toma de señales, entrenamiento de los clasificadores y prueba del modelo, los resultados se presentaron en la Sección de 1.3.

## 2.4 Pruebas realizadas a las señales adquiridas con la interfaz de usuario

Con el fin de comprobar la precisión de las predicciones de los clasificadores entrenados con señales adquiridas, se efectuó la siguiente prueba: a través de la interfaz gráfica

y sus respectivas etapas descritas en la Visualización, se entrenaron los clasificadores KNN y ANN con dos sujetos sanos de 23 años. Se realizaron una serie de 10 predicciones por cada uno de los 3 movimientos así fue posible calcular que tan exactas fueron las predicciones de los modelos, estos resultados se presentaron en la Sección 3.4.

## 3. RESULTADOS Y ANÁLISIS

### 3.1 Resultados de la identificación del conjunto de características

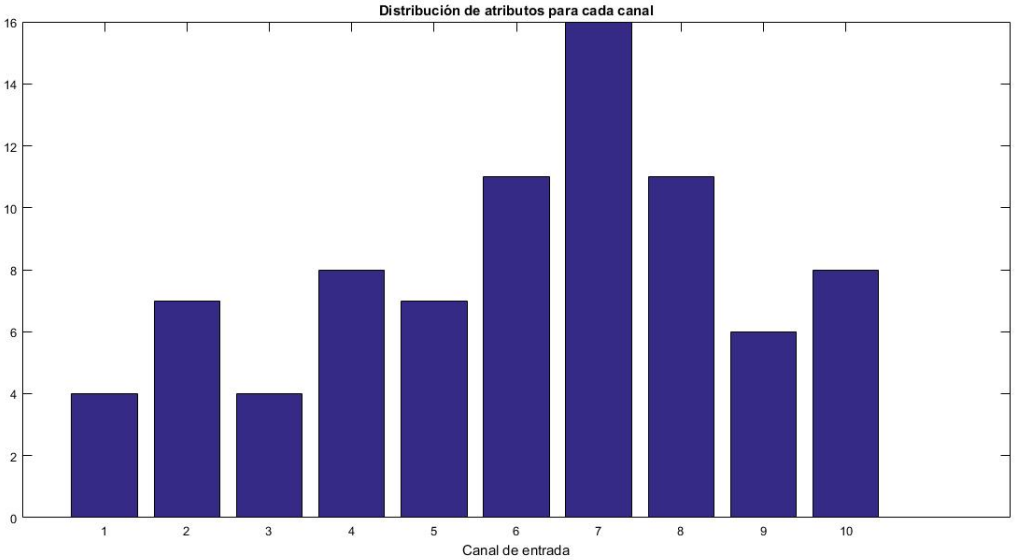
Por medio de las tablas y figuras se exponen los resultados de los experimentos mencionados en la sección anterior.

**Resultado experimento 1:** El algoritmo Cfs Subset Eval arrojó 82 características seleccionadas como las mejores del conjunto de 450. Se observa una distribución constante de la información a lo largo de los 10 electrodos y la moda en el canal 7. Permitiendo observar que cada canal pudo aportar información significativa a la clasificación. Se mostró en la **Figura 20** la distribución de frecuencias por electrodo.

Según este algoritmo de selección de atributos, la información más relevante perteneció a: frecuencia media modificada de la señal, desviación estándar y mediana de la señal en tiempo discreto, el valor rms de los primeros coeficientes de detalle de la transformada Wavelet y la varianza del tercer coeficiente de aproximación de la descomposición Wavelet.

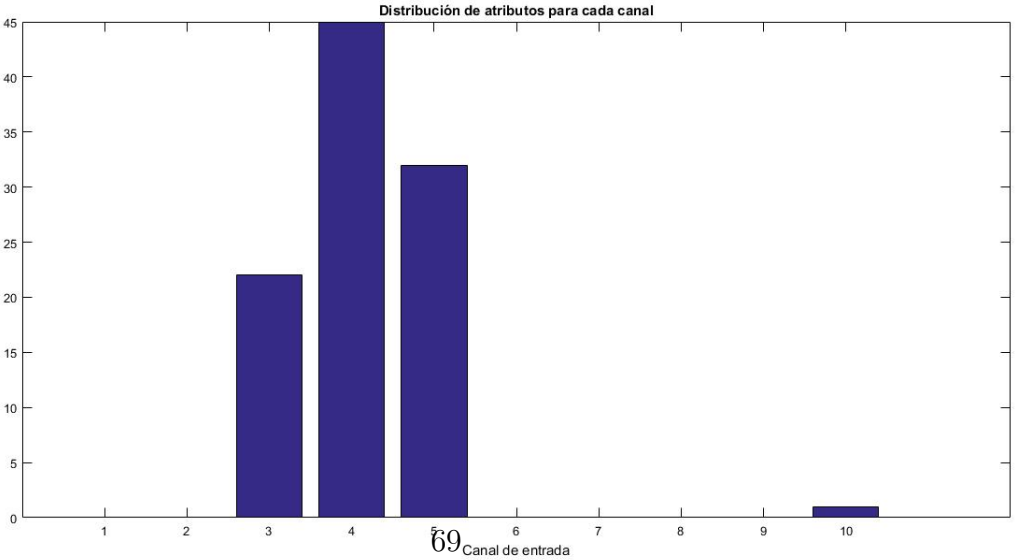
**Resultado experimento 2:** Se seleccionaron las mejores 100 características arrojadas por el método Classifier Attribute Eval. El método no arrojó los pesos para determinar la relevancia de cada uno. Y según la **Figura 21** tomó como información relevante los atributos de únicamente 3 canales y una mínima parte del canal 10. Según este algoritmo de selección de atributos, entre los primeros 100, la información más relevante perteneció a todo el conjunto de características, enfocándose principalmente en los coeficientes de curtosis, Fischer, rms varianza y entropía de la transformada Wavelet.

Figura 20. Histograma del vector de atributos obtenido con Cfs Subset Eval.



Fuente: Creación propia.

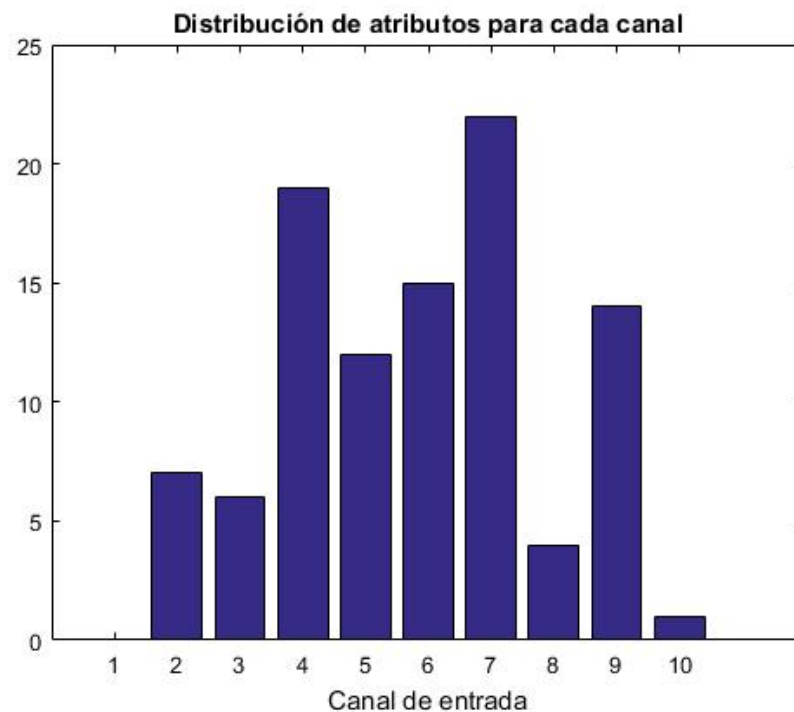
Figura 21. Histograma del vector de atributos obtenido con Classifier Attribute Eval.



**Resultado experimento 3:** Nuevamente al igual que en el experimento 1, se mostró en **Figura 22** una mayor concentración de información en el canal 7, perteneciente al músculo braquioradial. Se tomó las primeras 100 características para realizar la siguiente distribución de frecuencias. También se reveló al canal 1 como poco aportante de información, debido a la ausencia de gran masa muscular en la posición del electrodo ubicado cerca del codo.

Según este algoritmo de selección de atributos, entre los primeros 100, la información más relevante perteneció a: valor RMS y desviación estándar de la señal en el tiempo discreto, frecuencia media, entropía de los segundos coeficientes de detalle de la transformada Wavelet.

**Figura 22.** Histograma de vector de atributos arrojado por Correlation Attribute Eval.

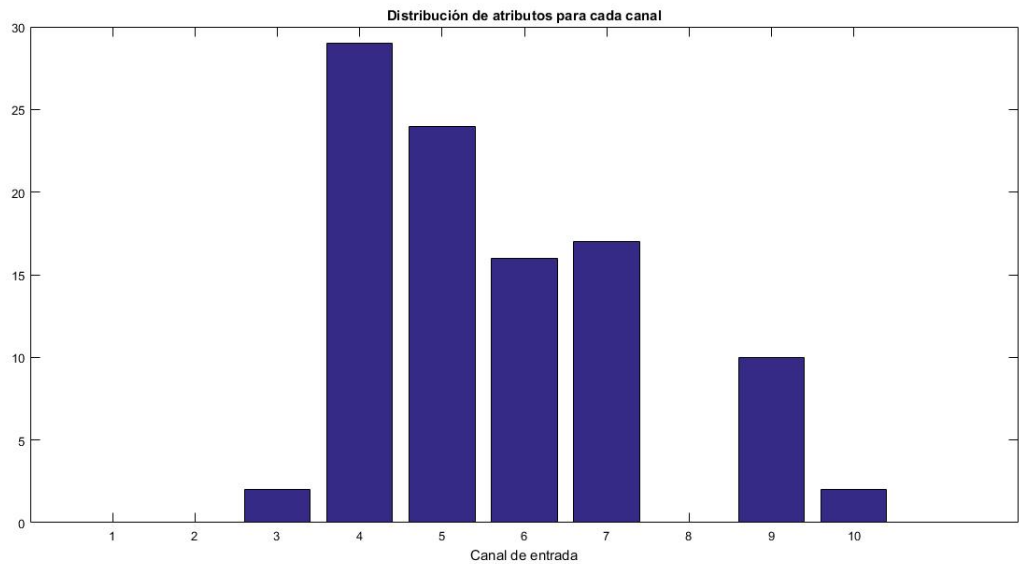


**Fuente:** Creación propia.

**Resultado experimento 4:** El algoritmo Gain Ratio Attribute Eval mostró en la **Figura 23** que 29 de las mejores 100 características están concentradas sobre el músculo flexor largo, y además que el 86% de la información relevante la aportaron los músculos de la parte superior del antebrazo, desde el flexor largo hasta el músculo braquioradial. Ignorando la información de los canales 1,2 y 8.

Según este algoritmo de selección de atributos, entre los primeros 100, la información más relevante perteneció a: desviación estándar y varianza de la señal en el tiempo, la integral bajo la curva, frecuencia media, coeficiente de Fischer y valor rms de los coeficientes de detalle del primer nivel de la transformada Wavelet.

**Figura 23. Histograma del vector de características arrojado por Gain Ratio Attribute Eval.**

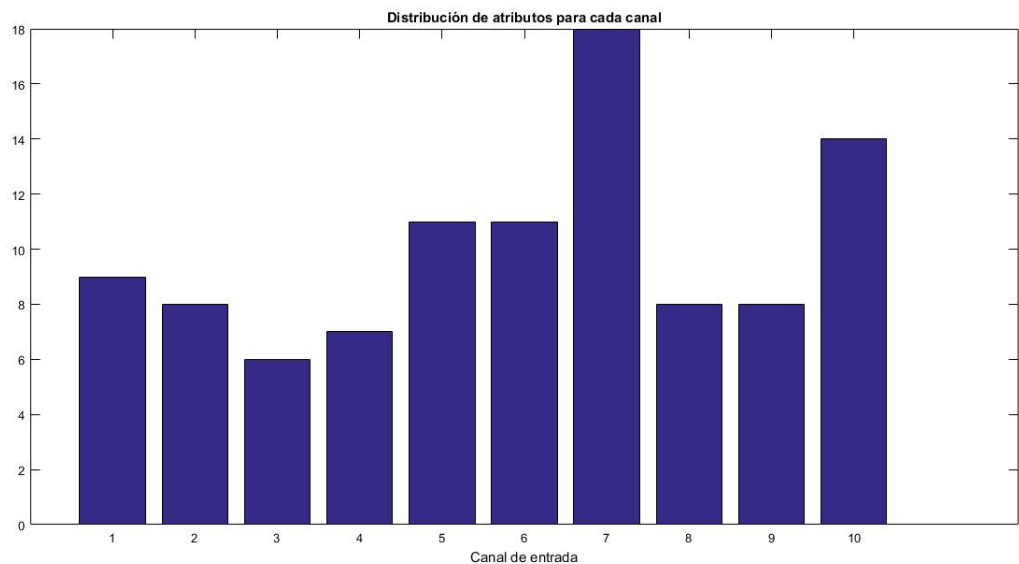


**Fuente:** Creación propia.

**Resultado experimento 5:** De los 5 métodos evaluados como se pudo observar en la gráfica **Figura 24**, el algoritmo Relief tomó entre los 100 mejores atributos información de todos los canales. Pero al igual que el experimento 3 y el experimento 1, la mayoría de la información se concentró en el canal número 7. En **Figura 24** se mostró la distribución de frecuencias de atributos.

Según este algoritmo de selección de atributos, entre los primeros 100, la información más relevante perteneció a: Desviación estándar, entropía y energía de Shannon de la señal en tiempo discreto, además de la amplitud de Willison, frecuencia media de la señal y la entropía de todos los coeficientes wavelet.

**Figura 24. Histograma del vector de características arrojado por Relief.**



**Fuente:** Creación propia.

Estos métodos de selección de características evidenció que existe un gran aporte de la

transformada *Wavelet* a la identificación de señales EMG, enfocándose principalmente en los coeficientes de detalle. También se observó que el canal 7 muestra gran concentración de la información en 3 diferentes métodos de selección de atributos. Para los 12 movimientos seleccionados, se dedujo que poseen gran actividad del músculo braquioradial y flexor largo y que su acción es significativamente diferenciable, en este conjunto de 12 movimientos.

### 3.2 Resultados de clasificación de señales EMG de Ninapro

A continuación se mostró los resultados de los experimentos descritos en la sección de Marco Experimental.

**Resultado experimento 6:** La **Figura 25** mostró en color rojo una curva correspondiente al porcentaje de error en función de la cantidad de características utilizadas. La curva superó el umbral del 10% de error de clasificación al utilizar más de 57 atributos de los arrojados por el algoritmo Cfs Subset Eval, y se obtuvo el mínimo clasificando con 82 características.

En la **Figura 26** se mostró la matriz de confusión obtenida del mejor modelo. El grupo de prueba correspondió a 81 muestras por cada clase. La clase con mejor número de muestras fue la clase 6, y con peor acierto de clasificación la clase 1, aún así, la clase 1 tuvo un acierto de clasificación de 91.4%.

Como resultado adicional se observó una tabla en la **Figura 27a** con la sensibilidad y especificidad promedio de las 10 repeticiones, con el mejor grupo de características y para cada clase. La sensibilidad para cada clase está por encima del 85%, e informó que el clasificador KNN es capaz de identificar con fidelidad los casos verdaderos de cada clase. Mientras que la especificidad indicó el grado de reconocer la no pertenencia a la clase de interés.



**Resultado experimento 7:** Con las características de Classifier Attribute Eval variando desde 1 hasta 100, se mostró en color magenta el porcentaje de error en la curva de la **Figura 25**. Se obtuvo una forma irregular (No solo decreciente como en el caso anterior) que pudo ser asociada a la información concentrada en su mayoría en solo 3 canales. Obteniendo un mínimo de 39% de error con 99 características.

En la **Figura 26** se registró la matriz de confusión del mejor clasificador de esta prueba. La clase con mayor cantidad de muestras clasificadas correctamente correspondió a la clase 8, y la de menor número de muestras clasificadas correctamente fue la clase 10 con tan solo 26 acertadas de 81 posibles. Esta información se complementó con la tabla en la **Figura 27b** de sensibilidad y especificidad, en donde se evidenció un resultado poco destacable del clasificador KNN. La única clase con sensibilidad por encima del 80% fue la clase 8.

**Resultado experimento 8:** Con características del algoritmo Correlation Attribute Eval, se observó en la **Figura 25** en color amarillo, una curva decreciente cuyo error se estabilizó a partir de los 70 atributos, con un mínimo de 8.84% utilizando 99 características.

La tabla **Figura 26** se consignó la matriz de confusión del mejor modelo KNN con 99 características y mostrando gran mejoría con respecto al experimento 7. Esta tabla mostró que la clase 3 y 12 presentan la mayor cantidad de muestras clasificadas correctamente. Y se repitió el resultado del experimento 7 de ser la clase 1 la de menor muestras clasificadas correctamente. La anterior información se respaldó con la de sensibilidad y especificidad promedio de las 10 repeticiones con KNN y 99 atributos, mostrada en la **Figura 27c**. Ambas medidas se redondearon a dos cifras decimales.

**Resultado experimento 9:** En la gráfica de error vs características (**Figura 25**) en color verde, se mostró una curva decreciente con un mínimo de 14,51% de error usando 97 características. Se evidenció también que el clasificador KNN no superó el

umbral del 10% de error para las primeras 100 características obtenidas con Gain Ratio Attribute Eval.

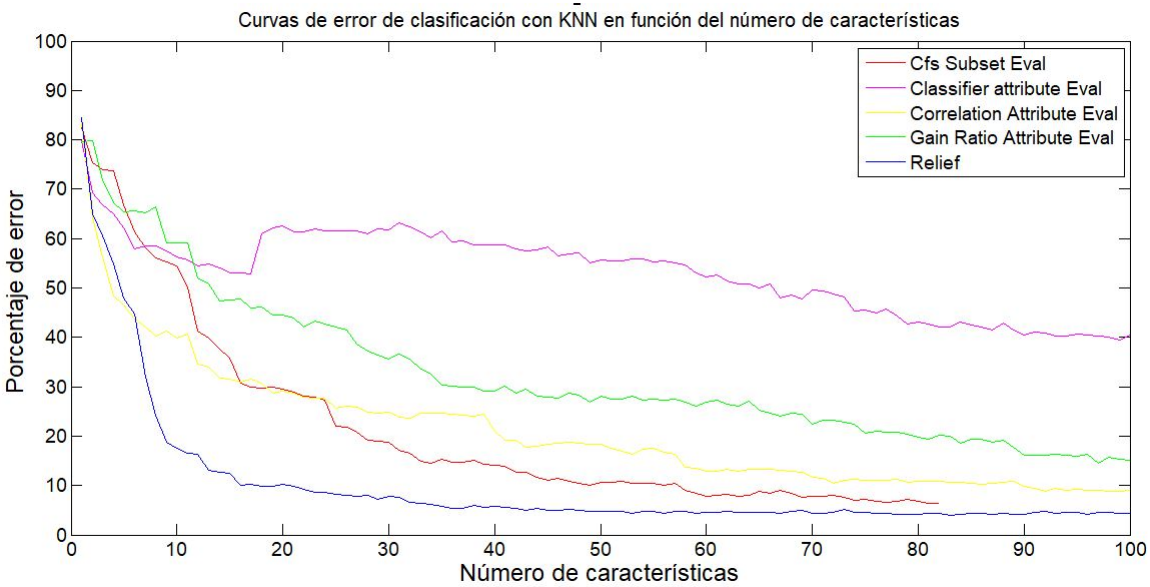
La matriz de confusión de la tabla **Figura 26** reveló que el bajo porcentaje de clasificación se debe nuevamente a la clase 1, 5 y a la clase 10, teniendo un porcentaje de clasificación acertada de 75.3%, 72,8% y 75,3% respectivamente. Mientras que en **Figura 27d** las clases con mayor cantidad de verdaderos positivos fueron la clase 6, clase 8 y clase 12.

**Resultado experimento 10:** El clasificador KNN con combinación con la variación de las características obtenidas con el algoritmo Relief, mostró en color azul la curva en la **Figura 25** con mayor tasa de decrecimiento a medida que se incrementaba la cantidad de atributos. Necesitando menor cantidad que los modelos anteriores, ya que superó el 10% de error con tan solo 20 características y se mantiene decreciendo de manera lenta hasta un mínimo de 4% con 83 atributos.

La matriz de confusión del mejor modelo se consignó en (**Figura 26**). Esta matriz reveló que todas las clases se clasificaron de manera uniforme. Sin embargo al igual que en los experimentos anteriores, a pesar de no ser una cantidad significativa, la clase 1 y 10 presentaron menor número de verdaderos positivos. la **Figura 27e** mostró varios valores de especificidad de 1, implicando una capacidad perfecta de identificar verdaderos negativos. Sin embargo hay que tener en cuenta que son valores promedio de diez pruebas y se redondeó ese valor, a dos cifras decimales.

**Resultado experimento 11:** El clasificador probabilístico Parzen mostró una curva de error, presentada en color rojo en la **Figura 28**, que decreció y alcanzó un buen porcentaje de clasificación con cerca de 60 atributos, con un mínimo de 5.85% usando 82 atributos. En la matriz de confusión presentada en la **Figura 29** se consignó los resultados de este experimento. Esta tabla mostró uniformidad en la clasificación de todas las clases, exceptuando la número 10. Presentó además según la **Figura 30a** la más baja sensibilidad para la clase 5.

Figura 25. Curvas de error de clasificación promedio del clasificador KNN, variando el número de características, para cada uno de los métodos de selección de atributos.



Fuente: Creación propia.

Figura 26. Matriz de confusión de los mejores modelos de clasificador de KNN, para los diferentes métodos de selección de características.

Clase Real	Predicción de los Mejores Modelos de KNN																							
	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7	Clase 8	Clase 9	Clase 10	Clase 11	Clase 12	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7	Clase 8	Clase 9	Clase 10	Clase 11	Clase 12
Clase 1	74	32	67	61	76	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Clase 2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Clase 3	0	3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Clase 4	0	9	1	2	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Clase 5	3	12	2	11	5	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Clase 6	2	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Clase 7	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Clase 8	1	1	0	2	1	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Clase 9	4	1	0	0	3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Clase 10	2	15	4	12	4	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Clase 11	0	1	1	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Clase 12	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
CFS Subset Eval	74	32	67	61	76	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Classfier Attribute Eval	74	32	67	61	76	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Correlaton Attribute Eval	74	32	67	61	76	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Gain Ratio Attribute Eval	74	32	67	61	76	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Relief	74	32	67	61	76	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
CFS Subset Eval	79	48	79	76	81	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Classfier Attribute Eval	79	48	79	76	81	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Correlaton Attribute Eval	79	48	79	76	81	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Gain Ratio Attribute Eval	79	48	79	76	81	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Relief	79	48	79	76	81	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
CFS Subset Eval	79	48	79	76	81	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Classfier Attribute Eval	79	48	79	76	81	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Correlaton Attribute Eval	79	48	79	76	81	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Gain Ratio Attribute Eval	79	48	79	76	81	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Relief	79	48	79	76	81	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
CFS Subset Eval	69	26	73	61	76	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Classfier Attribute Eval	69	26	73	61	76	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Correlaton Attribute Eval	69	26	73	61	76	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Gain Ratio Attribute Eval	69	26	73	61	76	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Relief	69	26	73	61	76	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
CFS Subset Eval	79	52	67	67	79	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Classfier Attribute Eval	79	52	67	67	79	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Correlaton Attribute Eval	79	52	67	67	79	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Gain Ratio Attribute Eval	79	52	67	67	79	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Relief	79	52	67	67	79	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
CFS Subset Eval	79	56	79	80	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Classfier Attribute Eval	79	56	79	80	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Correlaton Attribute Eval	79	56	79	80	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Gain Ratio Attribute Eval	79	56	79	80	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Relief	79	56	79	80	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

**Figura 27. Tablas de sensibilidad y especificidad promedio obtenidas de los experimentos: 6 - 10.**

(a) Tabla de sensibilidad y especificidad promedio del mejor conjunto de características de CFS Subset Eval, en combinación con clasificador KNN

	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7	Clase 8	Clase 9	Clase 10	Clase 11	Clase 12
Sensibilidad	0,89	0,96	0,97	0,96	0,88	0,98	0,95	0,96	0,95	0,87	0,92	0,97
Especificidad	0,99	1,00	0,99	1,00	0,99	1,00	1,00	1,00	0,99	0,99	0,99	1,00

(b) Tabla de sensibilidad y especificidad promedio del mejor conjunto de características de Classifier Attribute Eval, en combinación con clasificador KNN

	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7	Clase 8	Clase 9	Clase 10	Clase 11	Clase 12
Sensibilidad	0,38	0,65	0,67	0,63	0,45	0,70	0,72	0,81	0,59	0,36	0,59	0,72
Especificidad	0,92	0,96	0,95	0,92	0,92	0,96	0,97	0,97	0,95	0,92	0,95	0,96

(c) Tabla de sensibilidad y especificidad promedio del mejor conjunto de características de Gain Ratio Attribute Eval, en combinación con clasificador KNN

	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7	Clase 8	Clase 9	Clase 10	Clase 11	Clase 12
Sensibilidad	0,87	0,93	0,96	0,96	0,84	0,96	0,91	0,94	0,90	0,84	0,88	0,96
Especificidad	0,98	0,99	0,99	0,99	0,99	0,99	0,99	1,00	0,99	0,99	0,99	0,99

(d) Tabla de sensibilidad y especificidad promedio del mejor conjunto de características de Correlation Attribute Eval, en combinación con clasificador KNN

	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7	Clase 8	Clase 9	Clase 10	Clase 11	Clase 12
Sensibilidad	0,73	0,84	0,94	0,90	0,76	0,92	0,88	0,94	0,89	0,66	0,86	0,95
Especificidad	0,97	0,99	0,99	0,99	0,98	0,99	0,99	0,99	0,99	0,97	0,99	0,99

(e) Tabla de sensibilidad y especificidad promedio del mejor conjunto de características de Relief, en combinación con clasificador KNN

	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7	Clase 8	Clase 9	Clase 10	Clase 11	Clase 12
Sensibilidad	0,95	0,96	0,99	0,99	0,90	0,99	0,95	0,99	0,94	0,93	0,95	0,97
Especificidad	0,99	1,00	1,00	1,00	0,99	1,00	1,00	1,00	1,00	0,99	1,00	1,00

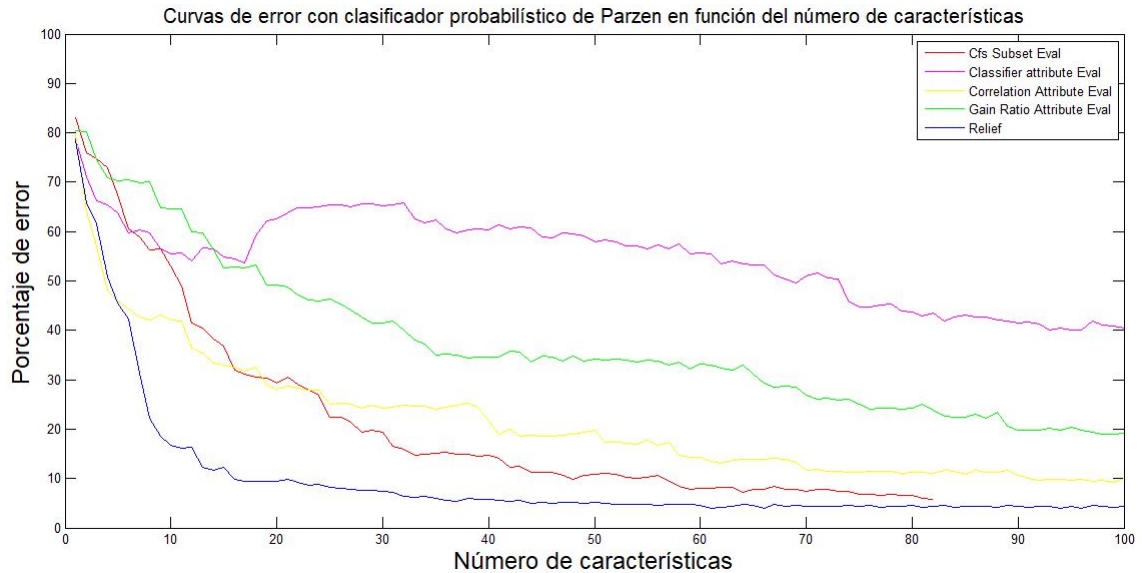
**Resultado experimento 12:** Se obtuvo una curva similar al experimento 7, mostrada en color magenta en la **Figura 28**. Además con un alto porcentaje de error, llegando a un mínimo de 40.1% para 96 características. La matriz de confusión en la **Figura 29** mostró nuevamente una muy baja eficiencia en las clases 1 y 10, ambas puestas de manera frecuente en la clase 4. Con la información de la **Figura 30b** se concluyó que con el clasificador parzen y el mejor conjunto de características la clase con mejor desempeño fue la número 8.

**Resultados experimento 13:** La **Figura 28** que mostró en color amarillo, el promedio de variación del error en función de la variación de cantidad de atributos, decreció hasta estabilizarse el promedio de error en alrededor de 10% con 70 atributos. Sin embargo el mínimo de 9,14% se obtuvo usando 99 de las 100 mejores características del algoritmo Correlation Attribute Eval. La matriz de confusión en la **Figura 29** mostró que, para este modelo que alcanzó el menor error, clasificó con baja precisión la clase 10. Sin embargo las altas medidas de las demás clases permitieron obtener un buen valor de especificidad promedio en la **Figura 30c**.

**Resultados experimento 14:** En esta prueba se obtuvo una curva que decreció lentamente y con un alto porcentaje de error, mostrada en color verde en la **Figura 28**. Su mejor modelo de clasificador alcanzó 18.89% de error usando 99 características, la matriz de confusión de la evaluación de este clasificador se mostró en la **Figura 29**. Además de obtener bajas medidas de sensibilidad para la clase 1, clase 5 y 10 que se mostraron en la **Figura 30d**. Y la clase con mejor porcentaje de clasificación fue la número 8 con 93,82%, y la peor fue la número 10 con 58% de acierto.

**Resultados experimento 15:** En esta prueba se obtuvo la curva de color azul de la **Figura 28**, resultando similar a la del experimento 10; en ambos experimentos se utilizó las características arrojadas por el algoritmo Relief. Se observó una curva que alcanza valores de 10% de error con tan solo 15 de los primeros 100 atributos. El mejor desempeño fue de 3,98% clasificando con 94 características. Sin embargo, se mantuvo

**Figura 28.** Curvas de error de clasificación promedio del clasificador probabilístico de Parzen, variando el número de características, para cada uno de los métodos de selección de atributos.



**Fuente:** Creación propia.

bastante constante desde los 40 atributos. La matriz de confusión en la **Figura 29** de esta prueba mostró una clasificación perfecta para todas las muestras de la clase 6 y 8. Mientras que la más baja con un 91,35% fue la clase 10, que clasificó correctamente 74 muestras de 81 posibles.

**Resultado experimento 16:** Hasta ahora en este experimento se mostró el mejor resultado de clasificación promedio con un 3,73% de error usando 81 atributos. Y con una curva que de creció rápido pero con pequeñas fluctuaciones en el promedio de error (ver la curva en color rojo en **Figura 31**). La matriz de confusión, consignada en la figura **Figura 32**, obtenida del mejor modelo de Red Neuronal, mostró un porcentaje de error de 3.5%, clasificando de manera perfecta las 81 muestras de la clase 9, mientras las segundas mejores fueron la número 5 y 11 con 80 correctas de 81.

Figura 29. Matriz de confusión de los mejores modelos de clasificador probabilístico de Parzen, para los diferentes métodos de selección de atributos.

Clase Real		Predicción de los Mejores Modelos del Clasificador Probabilístico de Parzen																																																																																																																						
Clase 1	70	24	69	50	77	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0																																																																																																		
Clase 2	0	0	0	3	0	78	49	75	63	77	0	5	0	2	0	0	0	0	0	0	0	0																																																																																																		
Clase 3	0	2	0	0	0	0	2	0	0	0	81	62	77	74	80	0	2	0	0	0	0	0																																																																																																		
Clase 4	1	7	0	0	0	0	0	0	0	0	2	0	0	0	77	58	80	68	79	0	5	0																																																																																																		
Clase 5	2	8	2	7	2	0	0	0	0	0	1	14	2	5	0	73	28	70	62	74	1	8																																																																																																		
Clase 6	0	3	0	0	0	0	0	0	0	0	0	1	0	81	56	76	68	81	0	0	0	1																																																																																																		
Clase 7	0	0	0	0	0	0	3	0	0	0	0	0	0	0	79	53	75	72	79	0	1																																																																																																			
Clase 8	0	0	0	0	0	0	4	0	0	0	0	0	0	0	78	69	77	76	81	0	1																																																																																																			
Clase 9	1	0	0	5	0	0	3	2	1	0	0	0	0	0	75	54	70	67	78	0	2																																																																																																			
Clase 10	6	18	8	6	2	0	0	0	0	0	0	2	15	1	6	0	6	11	4	21	4	1																																																																																																		
Clase 11	0	1	4	5	1	0	0	0	0	0	0	7	19	6	7	0	0	0	0	2	1	1																																																																																																		
Clase 12	0	0	0	1	0	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	3	0																																																																																																		
Clase 1	Clase 1	CFS Subset Eval	Clase 1	Classifier Attribute Eval	Clase 1	Correlation Attribute Eval	Clase 1	Gain Ratio Attribute Eval	Clase 1	Relief	Clase 2	CFS Subset Eval	Clase 2	Classifier Attribute Eval	Clase 2	Correlation Attribute Eval	Clase 2	Gain Ratio Attribute Eval	Clase 2	Relief	Clase 3	CFS Subset Eval	Clase 3	Classifier Attribute Eval	Clase 3	Correlation Attribute Eval	Clase 3	Gain Ratio Attribute Eval	Clase 3	Relief	Clase 4	CFS Subset Eval	Clase 4	Classifier Attribute Eval	Clase 4	Correlation Attribute Eval	Clase 4	Gain Ratio Attribute Eval	Clase 4	Relief	Clase 5	CFS Subset Eval	Clase 5	Classifier Attribute Eval	Clase 5	Correlation Attribute Eval	Clase 5	Gain Ratio Attribute Eval	Clase 5	Relief	Clase 6	CFS Subset Eval	Clase 6	Classifier Attribute Eval	Clase 6	Correlation Attribute Eval	Clase 6	Gain Ratio Attribute Eval	Clase 6	Relief	Clase 7	CFS Subset Eval	Clase 7	Classifier Attribute Eval	Clase 7	Correlation Attribute Eval	Clase 7	Gain Ratio Attribute Eval	Clase 7	Relief	Clase 8	CFS Subset Eval	Clase 8	Classifier Attribute Eval	Clase 8	Correlation Attribute Eval	Clase 8	Gain Ratio Attribute Eval	Clase 8	Relief	Clase 9	CFS Subset Eval	Clase 9	Classifier Attribute Eval	Clase 9	Correlation Attribute Eval	Clase 9	Gain Ratio Attribute Eval	Clase 9	Relief	Clase 10	CFS Subset Eval	Clase 10	Classifier Attribute Eval	Clase 10	Correlation Attribute Eval	Clase 10	Gain Ratio Attribute Eval	Clase 10	Relief	Clase 11	CFS Subset Eval	Clase 11	Classifier Attribute Eval	Clase 11	Correlation Attribute Eval	Clase 11	Gain Ratio Attribute Eval	Clase 11	Relief	Clase 12	CFS Subset Eval	Clase 12	Classifier Attribute Eval	Clase 12	Correlation Attribute Eval	Clase 12	Gain Ratio Attribute Eval	Clase 12	Relief



**Figura 30. Tablas de sensibilidad y especificidad promedio obtenidas de los experimentos: 11 - 15.**

(a) Tabla de sensibilidad y especificidad promedio del mejor conjunto de características de CFS Subset Eval, en combinación con clasificador KNN

	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7	Clase 8	Clase 9	Clase 10	Clase 11	Clase 12
Sensibilidad	0,91	0,96	0,97	0,98	0,87	0,98	0,96	0,97	0,93	0,88	0,92	0,98
Especificidad	0,99	1,00	0,99	1,00	0,99	1,00	1,00	1,00	1,00	0,99	1,00	1,00

(b) Tabla de sensibilidad y especificidad promedio del mejor conjunto de características de Classifier Attribute Eval, en combinación con clasificador KNN

	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7	Clase 8	Clase 9	Clase 10	Clase 11	Clase 12
Sensibilidad	0,29	0,59	0,75	0,74	0,38	0,79	0,70	0,83	0,57	0,28	0,67	0,60
Especificidad	0,95	0,98	0,90	0,90	0,94	0,90	0,98	0,97	0,98	0,94	0,92	0,97

(c) Tabla de sensibilidad y especificidad promedio del mejor conjunto de características de Gain Ratio Attribute Eval, en combinación con clasificador KNN

	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7	Clase 8	Clase 9	Clase 10	Clase 11	Clase 12
Sensibilidad	0,86	0,91	0,95	0,96	0,88	0,95	0,91	0,95	0,91	0,80	0,87	0,95
Especificidad	0,98	1,00	0,99	0,99	0,98	0,99	0,99	0,99	0,99	0,98	0,99	1,00

(d) Tabla de sensibilidad y especificidad promedio del mejor conjunto de características de Correlation Attribute Eval, en combinación con clasificador KNN

	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7	Clase 8	Clase 9	Clase 10	Clase 11	Clase 12
Sensibilidad	0,60	0,85	0,92	0,82	0,76	0,90	0,89	0,93	0,87	0,54	0,75	0,90
Especificidad	0,96	0,99	0,99	0,99	0,94	0,99	0,99	0,99	0,99	0,95	0,99	0,99

(e) Tabla de sensibilidad y especificidad promedio del mejor conjunto de características de Relief, en combinación con clasificador KNN

	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7	Clase 8	Clase 9	Clase 10	Clase 11	Clase 12
Sensibilidad	0,94	0,96	0,98	0,99	0,90	0,99	0,96	1,00	0,96	0,92	0,94	0,97
Especificidad	0,99	1,00	1,00	1,00	0,99	1,00	1,00	1,00	1,00	0,99	1,00	1,00

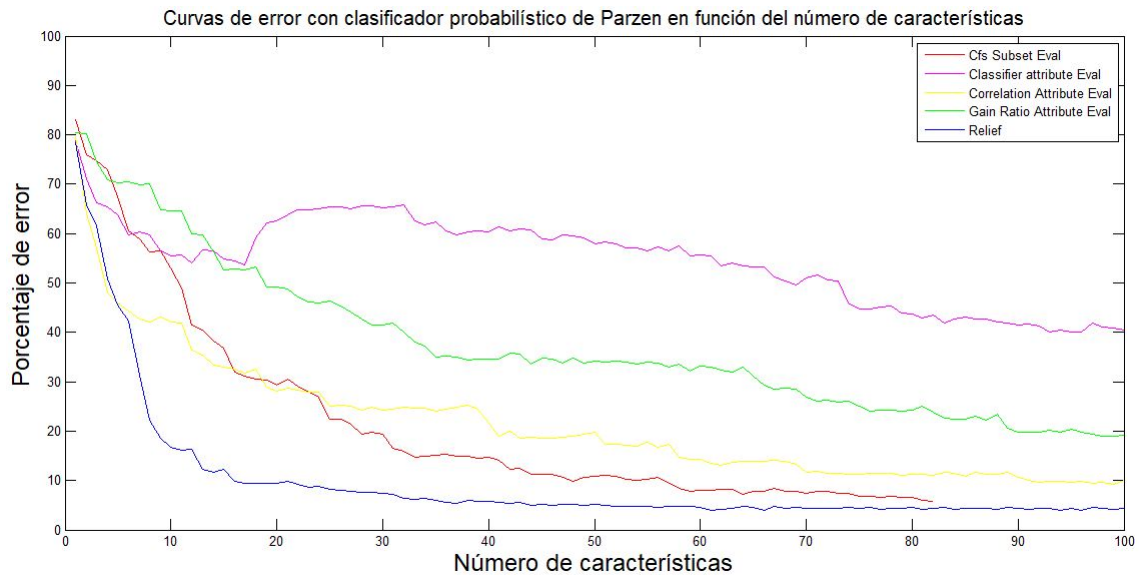
**Resultado experimento 17:** Con 99 atributos la Red Neuronal Artificial, obtuvo un mínimo de porcentaje promedio de error de 8,78%. La curva en color magenta en la **Figura 31**, presentó fluctuaciones similares a las vistas en el experimento 16. Se observó también que la gráfica decreció de manera lenta. La matriz de confusión (**Figura 32**) del mejor modelo, tuvo como mejor clase a la número 4, llegando a un 97,53% de acierto de clasificación. Sin embargo en general presentó un desempeño más bajo que la red del experimento 16.

**Resultados experimento 18:** La **Figura 31** en color amarillo, obtenida tras este experimento mostró un rápido descenso del error, al punto de obtener el 10% de error con solo las primeras 12 características de las arrojadas por el algoritmo Correlation Attribute Eval. Se obtuvo el mejor promedio de clasificación con 4,28% de error usando 91 características. Y la mejor red neuronal obtenida llega a un 3,81% de error, siendo su mejor clase la número 4, para la cual se clasificaron correctamente todas las muestras como se observó en la matriz de confusión en **Figura 32**.

**Resultados experimento 19:** Nuevamente el modelo de red neuronal presentó leves variaciones en la curva del error vs características (ver curva en color verde en **Figura 31**). Bajó del 10% de error, usando más de 26 atributos. Sin embargo, no mostró gran mejoría aumentando los atributos hasta el conjunto completo de 100 atributos. El error promedio mínimo ocurrió con 93 atributos llegando a 5,96%. La matriz de confusión en la **Figura 32** del mejor modelo de red neuronal mostró que clasificó las muestras de las clases 2, 8 y 12 por encima de un 98% de acierto. La tabla en **Figura 33d** mostró que las clases cuyos verdaderos positivos tuvieron mayor acierto fueron la clase 2 y la clase 11.

**Resultados experimento 20:** Para esta prueba se obtuvo curvas similares a las del experimento 15 y 10. En la curva de color azul en la **Figura 31**, se observó que el error decrece rápido con menores variaciones que las de los experimentos 16 a 19. Además, necesitó menor número de características que cualquier modelo para

**Figura 31.** Curvas de error de clasificación promedio de la Red Neuronal Artificial con Backpropagation, variando el número de características, para cada uno de los métodos de selección de atributos.



**Fuente:** Creación propia.

bajar de un porcentaje de error de 10%, con tan solo 10 atributos. Consiguió el mejor promedio de todos los clasificadores hasta ahora, obteniendo un 2,77% de error al usar 88 características. En promedio se clasificaron correctamente 945 muestras de 972 posibles, . Del mejor clasificador se obtuvo la matriz de confusión consignada en la **Figura 32**. En esta matriz se observó que, las clases con mayor cantidad de aciertos fueron la número 1,3,7 y 9 con 80 muestras bien clasificadas de 81 posibles. La más baja fue la clase 10 con 77 muestras de 81 posibles para un porcentaje de 95,06%. La información de los resultados obtenidos se resumió en la figura **Figura 34**.



**Figura 33. Tablas de sensibilidad y especificidad promedio obtenidas de los experimentos: 16 - 20.**

(a) Tabla de sensibilidad y especificidad promedio del mejor conjunto de características de CFS Subset Eval, en combinación con clasificador KNN

	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7	Clase 8	Clase 9	Clase 10	Clase 11	Clase 12
Sensibilidad	0,91	0,96	0,97	0,98	0,87	0,98	0,96	0,97	0,93	0,88	0,92	0,98
Especificidad	0,99	1,00	0,99	1,00	0,99	1,00	1,00	1,00	1,00	0,99	1,00	1,00

(b) Tabla de sensibilidad y especificidad promedio del mejor conjunto de características de Classifier Attribute Eval, en combinación con clasificador KNN

	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7	Clase 8	Clase 9	Clase 10	Clase 11	Clase 12
Sensibilidad	0,29	0,59	0,75	0,74	0,38	0,79	0,70	0,83	0,57	0,28	0,67	0,60
Especificidad	0,95	0,98	0,90	0,90	0,94	0,90	0,98	0,97	0,98	0,94	0,92	0,97

(c) Tabla de sensibilidad y especificidad promedio del mejor conjunto de características de Gain Ratio Attribute Eval, en combinación con clasificador KNN

	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7	Clase 8	Clase 9	Clase 10	Clase 11	Clase 12
Sensibilidad	0,86	0,91	0,95	0,96	0,88	0,95	0,91	0,95	0,91	0,80	0,87	0,95
Especificidad	0,98	1,00	0,99	0,99	0,98	0,99	0,99	0,99	0,99	0,98	0,99	1,00

(d) Tabla de sensibilidad y especificidad promedio del mejor conjunto de características de Correlation Attribute Eval, en combinación con clasificador KNN

	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7	Clase 8	Clase 9	Clase 10	Clase 11	Clase 12
Sensibilidad	0,60	0,85	0,92	0,82	0,76	0,90	0,89	0,93	0,87	0,54	0,75	0,90
Especificidad	0,96	0,99	0,99	0,99	0,94	0,99	0,99	0,99	0,99	0,95	0,99	0,99

(e) Tabla de sensibilidad y especificidad promedio del mejor conjunto de características de Relief, en combinación con clasificador KNN

	Clase 1	Clase 2	Clase 3	Clase 4	Clase 5	Clase 6	Clase 7	Clase 8	Clase 9	Clase 10	Clase 11	Clase 12
Sensibilidad	0,94	0,96	0,98	0,99	0,90	0,99	0,96	1,00	0,96	0,92	0,94	0,97
Especificidad	0,99	1,00	1,00	1,00	0,99	1,00	1,00	1,00	1,00	0,99	1,00	1,00

**Figura 34. Resumen de los resultados**

Experimento	Clasificador	Método de selección de atributos	Porcentaje de error mínimo	Número de características usadas
6	KNN	CFSSUBSETEVAL	6,3 ± 0,82	82
7	KNN	CLASSIFIERATTRIBUTEVAL	39,44 ± 1,36	99
8	KNN	CORRELATIONATTRIBUTEVAL	8,84 ± 0,64	99
9	KNN	GAINRATIOATTRIBUTEVAL	14,51 ± 01,14	97
<b>10</b>	<b>KNN</b>	<b>RELIEF</b>	<b>4,0 ± 0,46</b>	<b>83</b>
11	Parzen	CFSSUBSETEVAL	5,85 ± 1,01	82
12	Parzen	CLASSIFIERATTRIBUTEVAL	40,09 ± 0,95	96
13	Parzen	CORRELATIONATTRIBUTEVAL	9,14 ± 1,26	99
14	Parzen	GAINRATIOATTRIBUTEVAL	18,89 ± 1,0	99
<b>15</b>	<b>Parzen</b>	<b>RELIEF</b>	<b>3,98 ± 0,54</b>	<b>94</b>
16	Red Neuronal Artificial	CFSSUBSETEVAL	3,73 ± 0,62	81
17	Red Neuronal Artificial	CLASSIFIERATTRIBUTEVAL	8,78 ± 0,65	99
18	Red Neuronal Artificial	CORRELATIONATTRIBUTEVAL	4,28 ± 0,40	91
19	Red Neuronal Artificial	GAINRATIOATTRIBUTEVAL	5,96 ± 0,48	93
<b>20</b>	<b>Red Neuronal Artificial</b>	<b>RELIEF</b>	<b>2,77 ± 0,29</b>	<b>88</b>

**Fuente:** Creación propia.

A partir de los resultados consignados en la figura **Figura 34**, se concluyó que el mejor método de selección de características es el algoritmo Relief, en combinación con los 3 métodos de clasificación. Además Con un modelo de Red Neuronal Artificial con BackPropagation fue posible identificar la intención de movimiento, a partir de señales electromiográficas, con un buen desempeño, para ser utilizado en aplicaciones, que requieran interfaces hombre máquina.

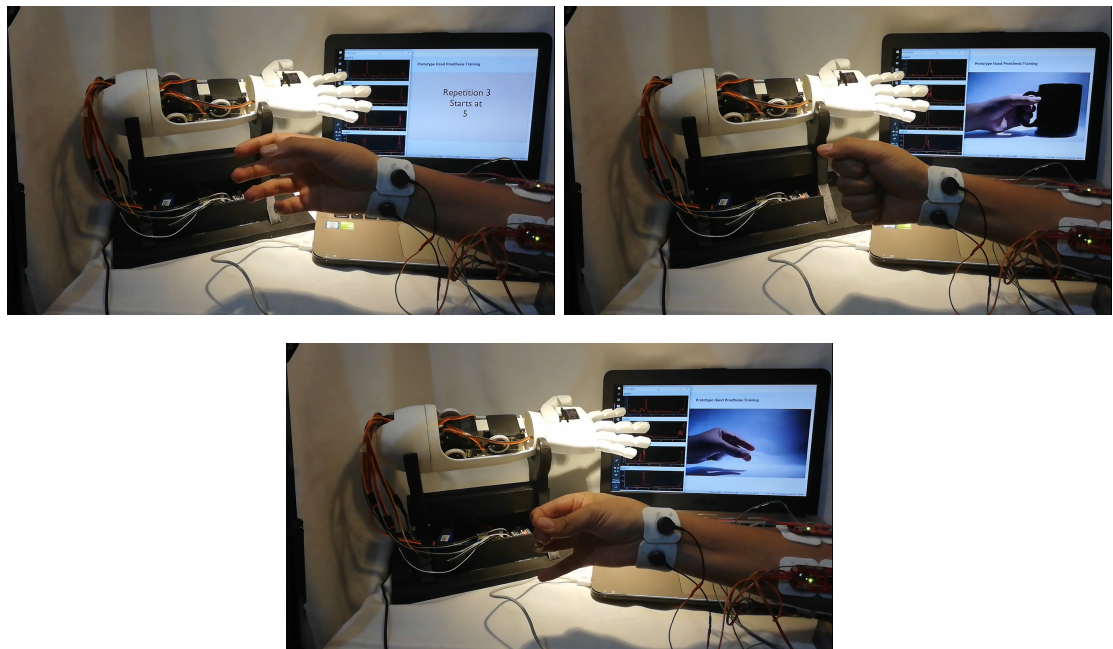
### 3.3 Resultados de la Interfaz gráfica y Actuador

Se presentó como resultados imágenes de todas las etapas de la interfaz gráfica descritas en la Sección ???. La etapa de *Inicio* se encuentra en la **Figura 35** donde se

tomó las señales EMG mientras el voluntario realizó los 3 movimientos, la etapa de *Entrenamiento* en la figura **Figura 36**, con las señales adquiridas se entrenaron los clasificadores ANN y KNN. La etapa de *Prueba de modelos* donde se obtuvo una nueva señal, luego se seleccionó el modelo para realizar la predicción y finalmente, la prótesis lo ejecutó, se mostró en la **Figura 37** con el movimiento 1, en la **Figura 38** con el movimiento 2 y en la **Figura 39** con el movimiento 3. Finalmente se presentó el actuador ejecutando los 3 movimientos, movimiento 1 en la **Figura 40**, movimiento 2 en **Figura 41** y movimiento 3 en **Figura 42**.

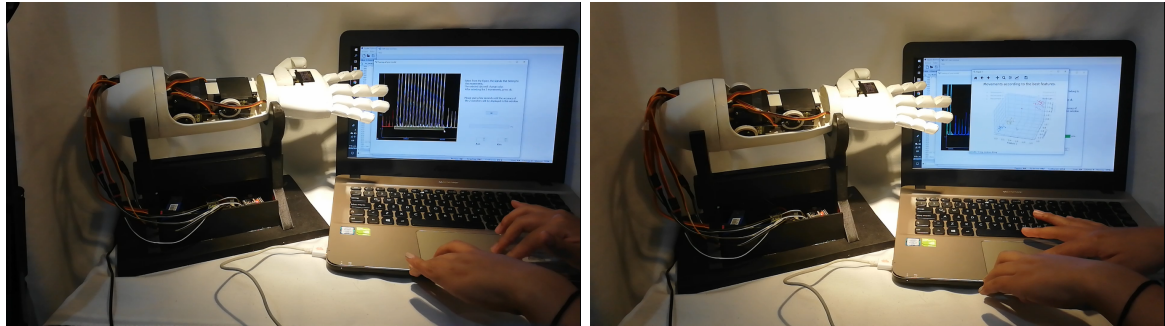
A través de las imágenes se pudo comprobar que la interfaz fue completamente funcional, cada una de sus etapas cumplió su función y además el actuador ejecutó los movimientos sin problema.

**Figura 35.** Ejecución de la etapa de *Inicio* de la interfaz gráfica.



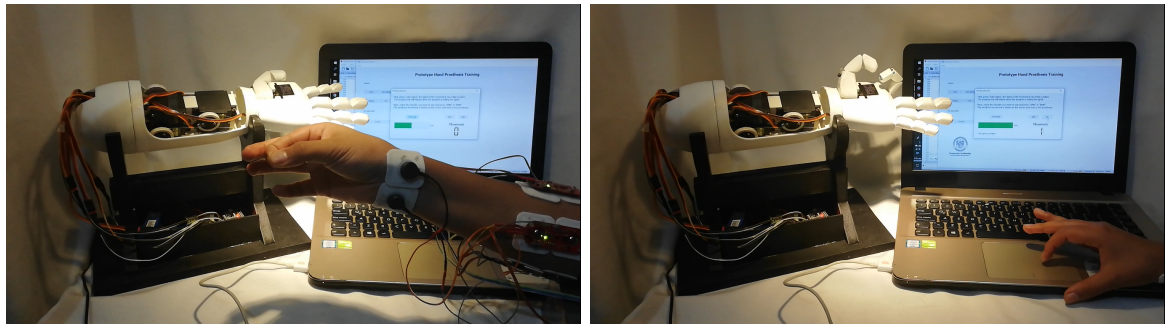
**Fuente:** Original de esta investigación.

Figura 36. Ejecución de la etapa de *Entrenamiento* de la interfaz gráfica.



Fuente: Original de esta investigación.

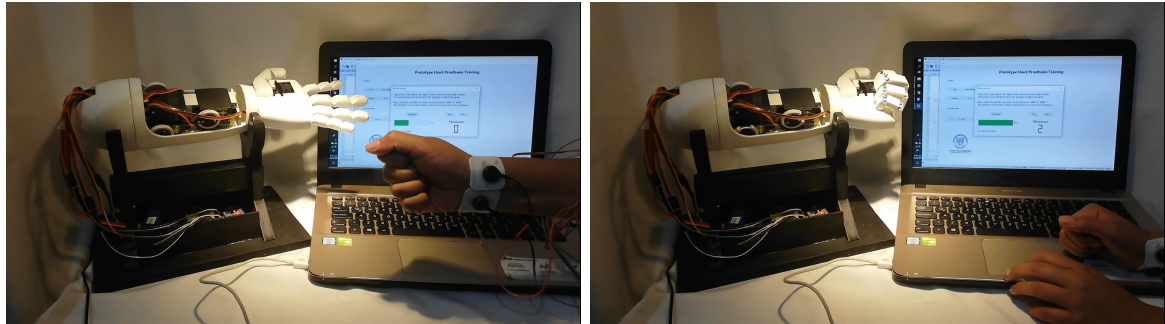
Figura 37. Ejecución de la etapa de *Prueba de los modelos* de la interfaz gráfica, movimiento 1.



Fuente: Original de esta investigación.

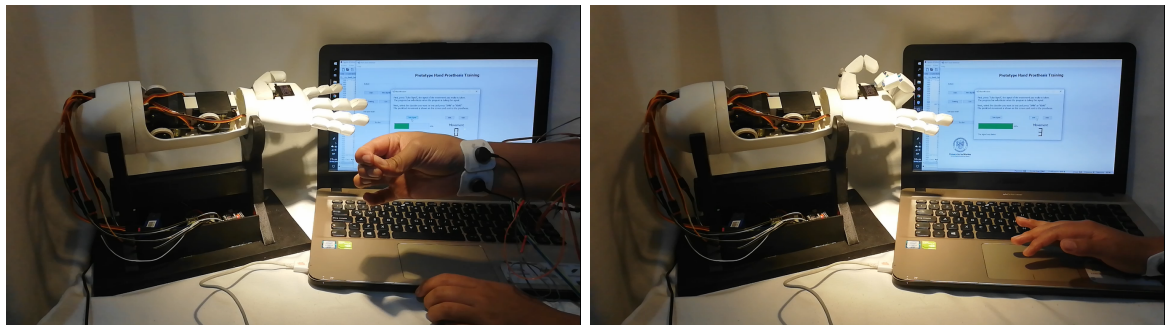


Figura 38. Ejecución de la etapa de *Prueba de los modelos* de la interfaz gráfica, movimiento 2.



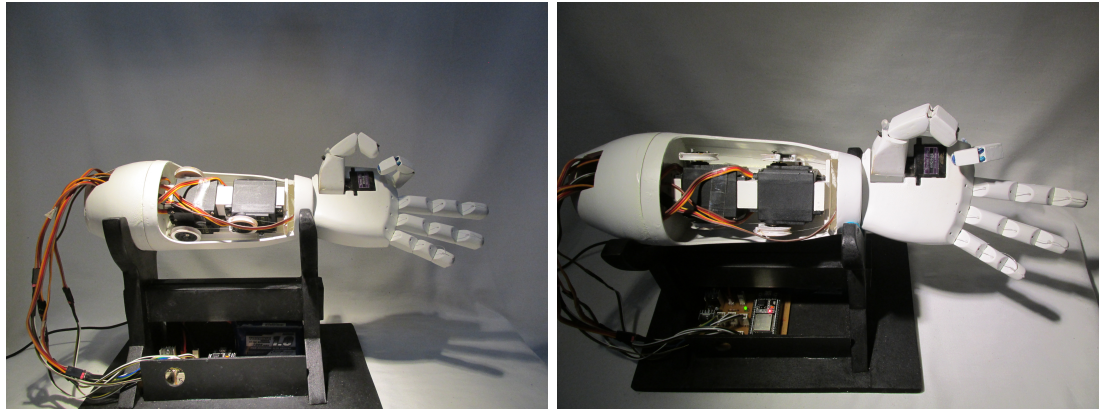
Fuente: Original de esta investigación.

Figura 39. Ejecución de la etapa de *Prueba de los modelos* de la interfaz gráfica, movimiento 3.



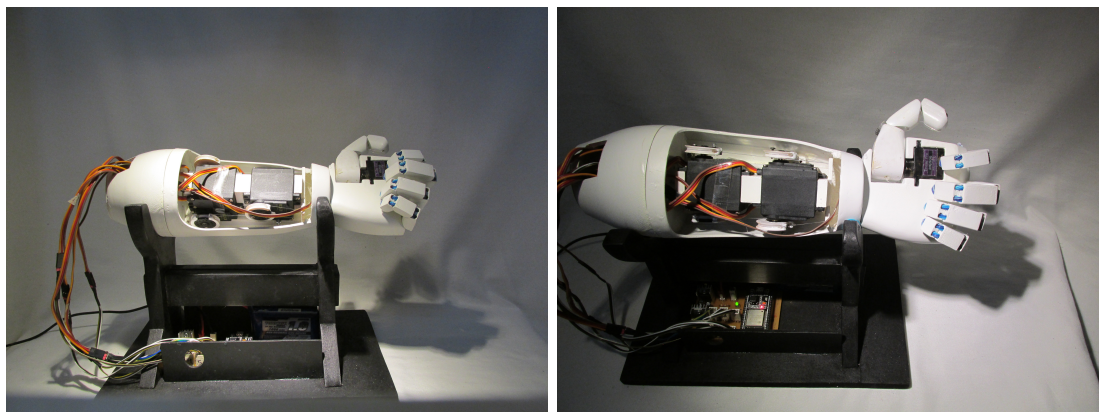
Fuente: Original de esta investigación.

**Figura 40.** Movimiento 1 ejecutado por el prototipo de prótesis.



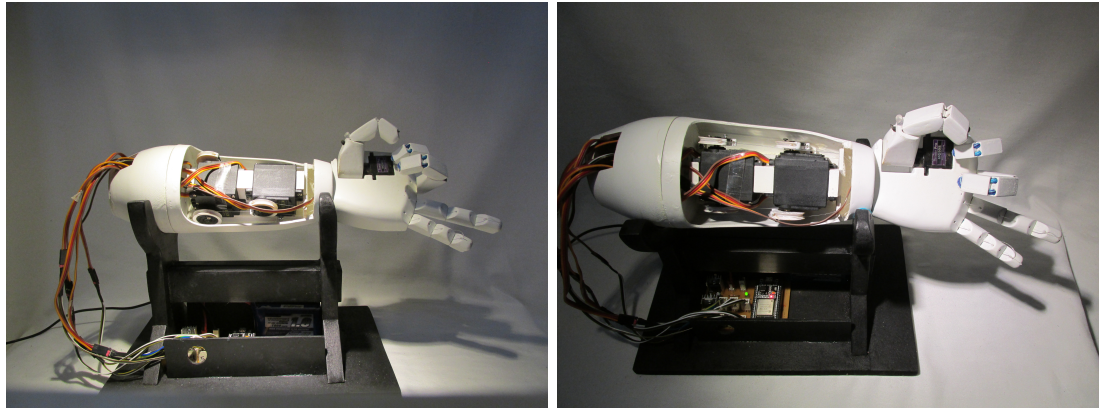
**Fuente:** Original de esta investigación.

**Figura 41.** Movimiento 2 ejecutado por el prototipo de prótesis.



**Fuente:** Original de esta investigación.

**Figura 42.** Movimiento 3 ejecutado por el prototipo de prótesis.



**Fuente:** Original de esta investigación.

### **3.4 Resultados de la prueba de modelos KNN y ANN con señales adquiridas**

Se presentó como resultado dos matrices de confusión de las 10 repeticiones individuales por cada uno de los 3 movimientos seleccionados e imágenes de las señales adquiridas de los 4 canales de cada sujeto. También los movimientos representados por las 3 mejores características como resultado de la interfaz.

**Sujeto 1:** El sujeto siguió el protocolo descrito en la Sección ???. El sujeto que realizó la prueba es un hombre de 23 años, contextura media, con una estatura 1.78 m y un peso de 75 Kg. Se encuentra que el modelo de KNN obtuvo un porcentaje de acierto de 86,6% y el modelo ANN obtuvo un 83,3% de acierto en su clasificación. Los resultados están en **Tabla 2** y **Tabla 3**. Se observó en ambos casos que el movimiento 3 es quien indujo los errores en las predicciones del modelo. En **Figura 43** se pudo observar las señales EMG del sujeto 1 ejecutando los 3 movimientos, para posteriormente realizar el entrenamiento de los clasificadores y probar la precisión de las predicciones. En **Figura 44**

se presentaron los 3 movimientos representados por las 3 mejores características según Relief.

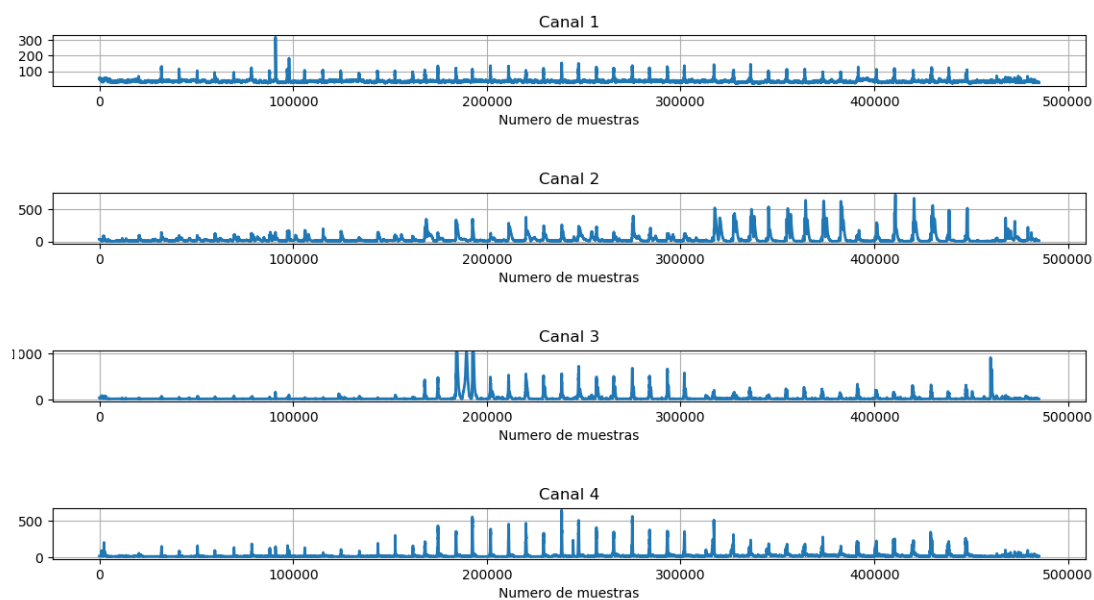
**Tabla 2. Matriz de confusión de la prueba de modelo KNN con sujeto 1**

		Predicción de modelo		
		Movimiento 1	Movimiento 2	Movimiento3
Clase real	Movimiento 1	<b>8</b>	0	2
	Movimiento 2	0	<b>8</b>	2
	Movimiento 3	0	0	<b>10</b>

**Tabla 3. Matriz de confusión de la prueba de modelo ANN con sujeto 1**

		Predicción de modelo		
		Movimiento 1	Movimiento 2	Movimiento3
Clase real	Movimiento 1	<b>7</b>	0	3
	Movimiento 2	0	<b>8</b>	2
	Movimiento 3	0	0	<b>10</b>

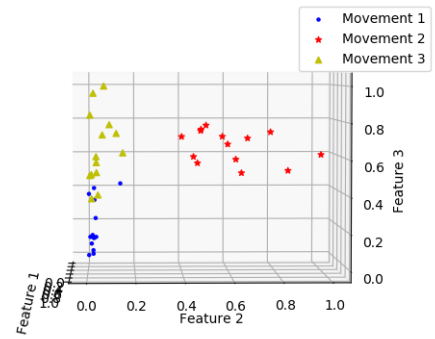
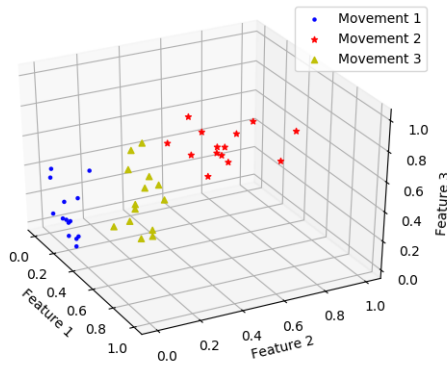
**Figura 43.** Señales EMG del sujeto 1 realizando los 3 movimientos para prueba de modelos KNN y ANN con señales adquiridas



**Fuente:** Creación propia.

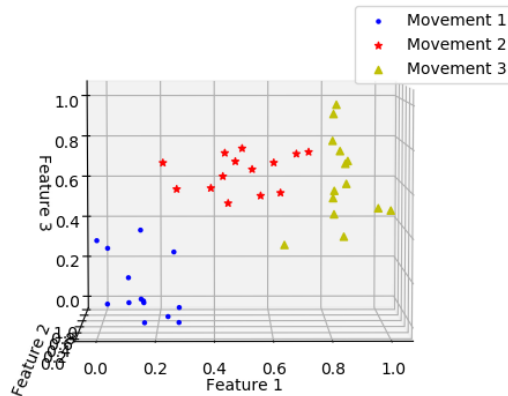
**Figura 44. Movimientos representado por las 3 mejores características del sujeto 1 para prueba de modelos KNN y ANN con señales adquiridas**

- (a) Vista general de las características. Movements according to the best features.
- (b) Característica 2 en el eje X y Característica 3 en el eje Y. Movements according to the best features.



- (c) Característica 1 en el eje X y Característica 3 en el eje Y.

Movements according to the best features.



**Fuente:** Original de esta investigación.

**Sujeto 2:** El sujeto siguió el protocolo descrito en la Sección ???. El sujeto que realizó la prueba es una mujer de 23 años, contextura media, con una estatura 1.58 m y un peso

de 55 Kg. Se encuentra que el modelo de KNN obtuvo un porcentaje de acierto de 80% y el modelo ANN obtuvo un 80% de acierto en su clasificación. Los resultados están en **Tabla 4** y **Tabla 5**. Se pudo resaltar que en ambos clasificadores, los movimientos 1 y 3 solían confundirse ya que tuvieron una ejecución muy parecida. En **Figura 45** se pudo observar las señales EMG del sujeto 2 ejecutando los 3 movimientos, para posteriormente realizar el entrenamiento de los clasificadores y probar la precisión de las predicciones. En **Figura 46** se presentó los 3 movimientos representados por las 3 mejores características según Relieff.

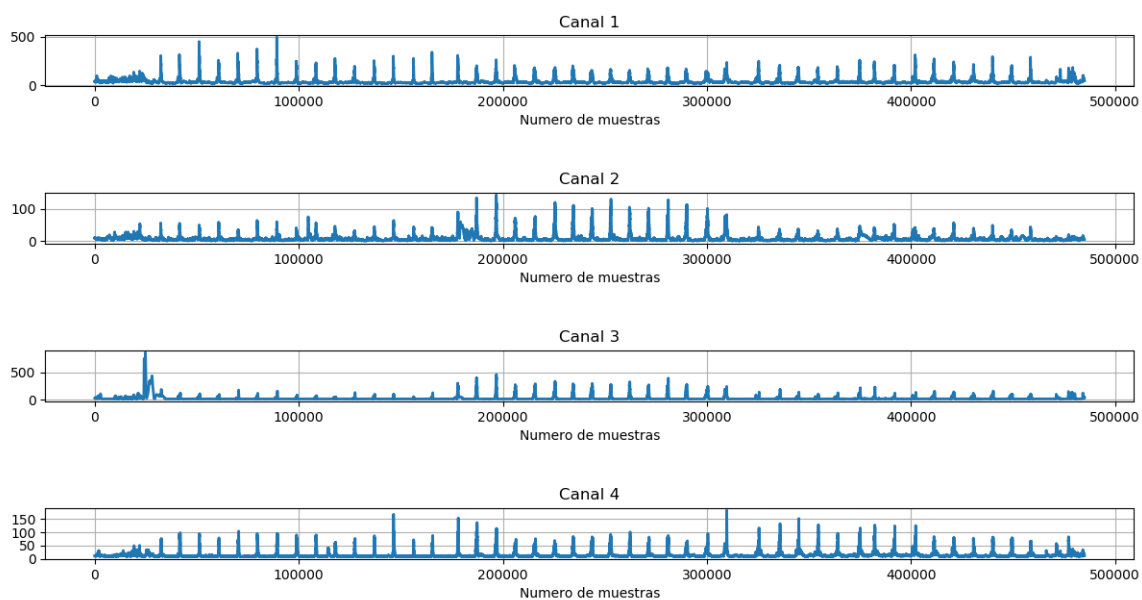
**Tabla 4. Matriz de confusión de la prueba de modelo KNN con sujeto 2.**

		Predicción de modelo		
		Movimiento 1	Movimiento 2	Movimiento3
Clase real	Movimiento 1	<b>6</b>	0	4
	Movimiento 2	0	<b>10</b>	0
	Movimiento 3	2	0	<b>8</b>

**Tabla 5. Matriz de confusión de la prueba de modelo ANN con sujeto 2.**

		Predicción de modelo		
		Movimiento 1	Movimiento 2	Movimiento3
Clase real	Movimiento 1	<b>9</b>	0	1
	Movimiento 2	2	<b>7</b>	1
	Movimiento 3	2	0	<b>8</b>

Figura 45. Señales EMG del sujeto 2 realizando los 3 movimientos para prueba de modelos KNN y ANN con señales adquiridas

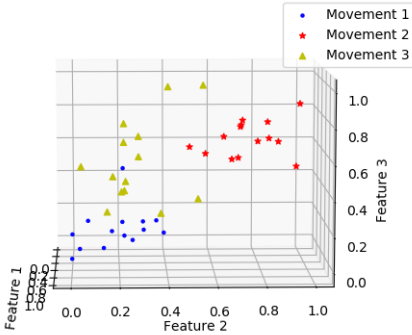
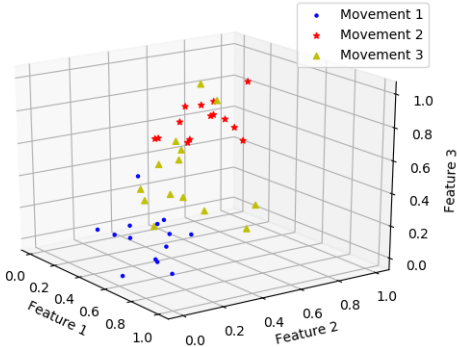


Fuente: Creación propia.



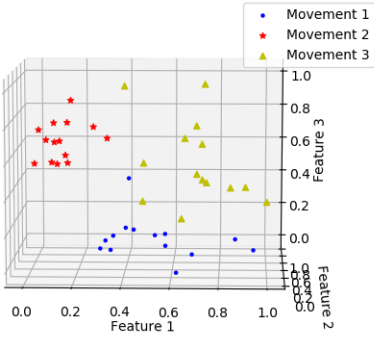
Figura 46. Movimientos representado por las 3 mejores características del sujeto 2 para prueba de modelos KNN y ANN con señales adquiridas

(a) Vista general de las características. Movements according to the best features.  
 (b) Característica 2 en el eje X y Característica 3 en el eje Y. Movements according to the best features.



(c) Característica 1 en el eje X y Característica 3 en el eje Y. Movements according to the best features.

Movements according to the best features.



Fuente: Original de esta investigación.

## 4. CONCLUSIONES

- Se comprobó que es posible identificar la intención de movimiento a partir de señales electromiográficas, tanto a partir de una base de datos como de señales en tiempo real y con un porcentaje de acierto considerablemente bueno (por encima del 95% con las condiciones adecuadas). Basado en el resultado del error medio y la desviación estándar, el clasificador ANN presenta un mejor desempeño que KNN y Parzen, sin embargo, todos los clasificadores tienen un porcentaje de error inferior al 10%. Estos resultados son comparables a los resultados obtenidos en (Phinyomark, 2011), donde se usa Redes Neuronales con Backpropagation se obtuvo un rendimiento del 98.21% pero clasificando solo 5 movimientos. Las tasas de reconocimiento fueron de 84.9% para el clasificador KNN en (Kim, 2011) donde se clasificaron cinco movimientos de muñeca, este proyecto destaca la dificultad de colocar los electrodos en el antebrazo, pero en nuestro caso, la base de datos utilizada se adquirió con un protocolo estricto para evitar este problema, esto hace una diferencia en los resultados obtenidos.

Con los resultados mencionados se puede considerar la posibilidad de ser utilizadas para el control de una prótesis de tipo electromecánica con control inteligente, que pueda ser utilizada por un paciente en su proceso de rehabilitación, recuperación de movilidad o para llevar a cabo tareas rutinarias. Además los resultados obtenidos permiten diseñar otro tipo de sistemas como una interfaz hombre-máquina para diferentes aplicaciones.

- Se determinó que existen un conjunto de movimientos que presentan mayor separabilidad. Diferenciándose especialmente aquellos con uso activo del músculo braquioradial. Esta información es soportada por el hecho de que los métodos de selección de atributos aplicados, arrojaron que los atributos más relevante pertenecen a los electrodos ubicados en esta sección del antebrazo.
- Los métodos de selección de atributos en combinación con la clasificación, arrojó que

a pesar de no detectar atributos redundantes, el algoritmo Relief es capaz de dar los mejores resultados, independientemente de la técnica de clasificación utilizada. Lo anterior teniendo como criterio el número de características necesarias para lograr un buen desempeño y el porcentaje de clasificación con dicha técnica. Además se determinó que los coeficientes wavelet aportan la información relevante para clasificar las señales EMG.

- El procedimiento de implementación de un sistema de detección de señales EMG en tiempo real, supone un reto diferente a trabajar con base de datos. Lo anterior debido a las complicaciones de la adquisición de señales, incluyendo sensores utilizados, posición del paciente, contacto con tierra, ruidos inherentes de componentes y red eléctrica, estado fisiológico del músculo, estado anímico del paciente, repetitividad en las ejecuciones de intención motora y posicionamiento de electrodos. Todas estas condiciones afectan el desarrollo de una prótesis portable, pero más sin embargo es posible identificar con gran fiabilidad una serie de movimientos elementales para ejecutar en la vida diaria.

## 5. TRABAJO FUTURO

Como trabajo futuro se propone utilizar la metodología propuesta en este trabajo, con una base de datos de señales EMG de personas amputadas, disponible también en Ninaweb, con el fin de comparar las señales, métodos de selección de características y movimientos y la eficiencia de los clasificadores, frente a estas nuevas señales. Realizar una búsqueda de características intrínsecas de señales EMG, esto se refiere a aquellas características que no se ven afectadas por factores externos, como son el estado físico de la persona, ruidos de la red eléctrica, adherencia de los electrodos, hidratación de la persona, entre otras. Mejorar el prototipo de prótesis, hacerlo totalmente autónoma, donde la predicción de los movimientos se realice sin necesidad de un equipo de cómputo.

# BIBLIOGRAFÍA

ALPADIN, Ethem. Introduction to Machine Learning. MIT Press, 2010.

AMBULODEGUI, Edwin. Manual de anatomía humana. Disponible en: <http://oncouasd.files.wordpress.com/2015/06/manualdeanatomiahumana.pdf>.

AROZI, Moh; PUTRI, Farika; ARIYANTO, Mochammad; CAESARENDRA, Wahyu; WIDYOTRIATMO, Augie; SETIAWAN, Joga; et al. Electromyography (emg) signal recognition using combined discrete wavelet transform based on artificial neural network (ann). In Industrial, Mechanical, Electrical, and Chemical Engineering, International Conference of 2016, IEEE, pp. 95-99.

ATZORI, Manfredo; GIJSBERTS, Arjan; CASTELLINI, Claudio; CAPUTO, Barbara; HAGER, Anne-Gabrielle; ELSIG, Simone; GIATSIDIS, Giorgio; BASSETTO, Franco y MULLER, Henning. Electromyography data for non-invasive naturally-controlled robotic hand prostheses. Scientific data 1, 2014.

ATZORI, Manfredo; GIJSBERTS, Arjan; KUZBORSKIJ, Ilja; ELSIG, Simone; HAGER, Anne-Gabrielle; DERIAZ, Olivier; CASTELLINI, Claudio; MULLER, Henning y CAPUTO, Barbara. Characterization of a benchmark database for myoelectric movement classification. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 2015.P.73-83.

BEJARANO, Sandra; SILVA, Fernanda y THOISON, Stephen. Desarrollo de un sistema de entrenamiento para el control de una prótesis mioeléctrica con un modelo de prótesis virtual. Memorias del XVIII Concurso Lasallista de Investigación, 2016.

BRAZEIRO, Jorge; PETRACCIA, Sabrina. Mano controlada por señales musculares. Master's thesis, Universidad de la República, Septiembre 2015.

CANAL, Mehmet. Comparison of wavelet and short time fourier transform methods in the analysis of emg signals. J Med Syst, 2008. P, 91-94.

CENE, Vinícius Horn y BALBINOT, Alexandre. Optimization of features to classify upper limb movements through sEMG signal processing. Brazilian Journal of Instrumentation and Control, 2016. P.14-20.

CIFUENTES, Irving Aaron. Diseño y construcción de un sistema para la detección de señales electromiográficas. UADY 1, 2010. P. 1-110.

DINAKARAN, S y THANGAIAH, Jeba. Role of attribute selection in classification algorithms. International Journal of Scientific Engineering Research, 2013.

GARCÍA, José. Comenzando con weka: Filtrado y selección de subconjuntos de atributos basada en su relevancia descriptiva para la clase, 2016.

HALL, Mark. Correlation-based feature selection for machine learning. Master's thesis, The University of Waikato, 1999.

KIM, Kang Soo; CHOI, Heung Ho; MOON, Choi Soo; y MUN, Chi Woong. Comparison of k-nearest neighbor, quadratic discriminant and linear discriminant analysis in classification of electromyogram signals based on the wrist-motion directions. Current Applied Physics, 2011. P 740 - 745.

LAENCINA, Pedro. Estimación de densidad de probabilidad mediante ventanas de parzen. Jornadas de Introducción a la Investigación en la UPCT, 2010.

LAVALLE, Manuel. Selección de Atributos Considerando Interdependencias. PhD tesis, Tecnológico de Monterrey, 2007.

MISIT, Michel; MISITI, Yves; OPPENHEIM, Georges y POGGI, Jean. Wavelet Toolbox™ 4, User's Guide. The MathWorks, Inc. disponible en : [https://www.ltu.se/cms\\_fs/1.51590!/wavelet%20toolbox%204%20user's%20guide%20\(larger%20selection\).pdf](https://www.ltu.se/cms_fs/1.51590!/wavelet%20toolbox%204%20user's%20guide%20(larger%20selection).pdf).

MOORE, Keith; DAILEY, Arthur y AGUR, Anne. Anatomía con orientación clínica. Lippincott Williams Wilkins, 2013.

MORALES, Ruben; MORALES, Diego y GRISALES, Victor. Caracterización de señales electromiográficas para la discriminación de seis movimientos de la mano. Scientia et Technica, 2009.

MOUJAHID, Abdelmalik; Inza, Iñaki y LARRAÑAGA, Pedro. Clasificadores k-nn. Tech. rep., Universidad del País Vasco-Euskal Herriko Unibertsitatea.

MOYANO, José. Instrumentación electrónica de comunicaciones. Tech. rep., ESCUELA TÉCNICA SUPERIOR DE INGENIEROS INDUSTRIALES Y DE TELECOMUNICACIÓN, 2005.

PAMBUDI, Mahfud Ridho; SIGIT, Riyanto; HARSONO, Tri. The bionic hand movement using myo sensor and neural networks. En 2016 International Conference on Knowledge Creation and Intelligent Computing (KCIC). IEEE, 2016. p. 259-264.

NAVARRO, Rafael. Electromiografía en Instrumentación Biomedica. Universidad de Alcalá.2016.

NEGI, Sachin; KUMAR, Yatindra; MISHRA, V. M. Feature extraction and classification for EMG signals using linear discriminant analysis. En 2016 2nd International Conference on Advances in Computing, Communication, & Automation (ICACCA)(Fall). IEEE, 2016. p. 1-6.

PHINYOMARK, Angkoon; PHUKPATTARANONT, Pornchai y LIMSAKUL, Chusak. A review of control methods for electric power wheelchairs based on electromyography signals with special emphasis on pattern recognition. IETE Technical Review, 2011. P.316-326.

PODRUG, Ermin y SUBASI, Abdulhamit. Surface emg pattern recognition by using dwt feature extraction and svm classifier. In The 1st Conference of Medical and Biological Engineering in Bosnia and Herzegovina, 2015.

PROAKIS, J. G y MANOLAKIS, Dimitris. Tratamiento Digital de Señales. Prentice Hall, 2007.

QUIROZ, F. A. R. Metodología de diseño de filtros digitales basada en estrategias de optimización metaheurística. Tesis de Maestría, Universidad Nacional de Colombia, 2017.

ROMO, Harold; REALPE, Judy y JOJOA, Pablo. Surface emg signals analysis and its applications in hand prosthesis control. Revista Avances en Sistemas e Informática, 2007.p 127-136.

SHI, Haijian. Best-first decision tree learning. Master's thesis, The University of Waikato, 2006.

URBANOWICZ, Ryan; MEEKER, Melissa; LACAVA, William; OLSON, Randal y MOORE, Jason. Reliefbased feature selection: Introduction and review. Journal of Biomedical Informatics, 2017.

VAZQUEZ, Sergio; OUBRAM, Outmane y ALI, Bassam. Intelligent recognition system of myoelectric signals of human hand movement. In International Symposium on Intelligent Computing Systems (2018), Springer, pp. 97-112.

YANNAKAKIS, Georgios y TOGELIUS, Julian. Artificial Intelligence and Games. Springer, 2018.

# ANEXOS



## **ANEXO A**

### **Artículo publicado en CCC 2018: Advances in Computing**

En la siguiente hoja se presenta el paper sometido al 13vo Congreso Colombiano de Computación (13CCC) desarrollado en Septiembre de 2018 en Cartagena de Indias, Colombia.

# Exploration of characterization and classification techniques for movement identification from EMG signals: Preliminary results

A. Viveros-Melo<sup>1</sup>, L. Lasso-Arciniegas<sup>2</sup>, J. A. Salazar-Castro<sup>3</sup>, D. H. Peluffo-Ordóñez<sup>4,5</sup>, M. A. Becerra<sup>6</sup>, A. E. Castro-Ospina<sup>7</sup>, and E. J. Revelo-Fuelagán<sup>8</sup>

<sup>1</sup>Universidad de Nariño, Colombia,

<sup>2</sup>Universidad de Nariño, Colombia,

<sup>3</sup>Corporación Universitaria Autónoma de Nariño, Colombia,

<sup>4</sup>Yachay Tech, Urcuqui, Ecuador,

<sup>5</sup>Corporación Universitaria Autónoma de Nariño, Colombia,

<sup>6</sup>Instituto tecnológico metropolitano, Colombia,

<sup>7</sup>Instituto tecnológico metropolitano, Colombia,

<sup>8</sup>Universidad de Nariño, Colombia.

**Abstract.** Today, human-computer interfaces are increasingly more often used and become necessary for human daily activities. Among some remarkable applications, we find: Wireless-computer controlling through hand movement, wheelchair directing/guiding with finger motions, and rehabilitation. Such applications are possible from the analysis of electromyographic (EMG) signals. Despite some research works have addressed this issue, the movement classification through EMG signals is still an open challenging issue to the scientific community -especially, because the controller performance depends not only on classifier but other aspects, namely: used features, movements to be classified, the considered feature-selection methods, and collected data. In this work, we propose an exploratory work on the characterization and classification techniques to identifying movements through EMG signals. We compare the performance of three classifiers (KNN, Parzen-density-based classifier and ANN) using spectral (Wavelets) and time-domain-based (statistical and morphological descriptors) features. Also, a methodology for movement selection is proposed. Results are comparable with those reported in literature, reaching classification errors of 5.18% (KNN), 14.7407% (ANN) and 5.17% (Parzen-density-based classifier).

**Keywords:** Classification, EMG signals, Movements selection, Wavelet.

## 1 Introduction

Electromyographic signals (EMG) are graphical recordings of the electrical activity produced by the skeletal muscles during movement. The analysis of EMG signals has traditionally been used in medical diagnostic procedures, and more

recently its applicability in human-machine control interfaces (HCIs) designing has increased, becoming -at some extent- indispensable for the activities of people's lives. Some remarkable applications of EMG-based HCI to mention are: Wireless-computer controlling through hand movement, wheelchair directing/guiding with finger motions, and rehabilitation [1, 2].

The muscle-signal-based control is possible thanks to the development of fields such as microprocessors, amplifiers, signal analysis, filtering and pattern recognition techniques. One of the main branches in the investigation of EMG signal recognition is that one aiming to identify features providing a better description of a specific movement. Often, such an identification process results in a difficult task since this kind of signals are sensitive to several artifacts, such as noise from electronic components, the action potentials that activate the muscles, the patient's health, physical condition, and hydration level, among others [3]. For this reason, it is essential to well-perform a preprocessing stage so that such artifacts can be corrected or mitigated and therefore a proper, cleaner EMG signal is obtained, being more suitable for any today's application, e.g. a prosthesis' control.

Along with adequate acquisition and preprocessing, EMG signals also require a characterization procedure consisting of extracting the most representative/informative, separable features and measures from the original signal so that the subsequent classification task may work well. That said, every signal EMG-signal processing stage plays a crucial role in the automatic movement identification [4–6]. Despite this research problem has a lot of manners to be addressed, it still lacks a definite solution and then remains a challenging, open issue.

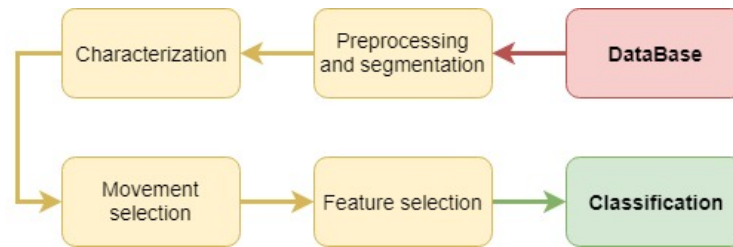
Consequently, in this work, we present an exploratory study on characterization and classification techniques to identify movements through EMG signals. In particular, the spectral features (wavelet coefficients), temporal features and statistics are used (Area under the curve, absolute mean value, effective value, standard deviation, variance, median, entropy) [1, 4, 7, 8]. The characterization of the signal, leaves a matrix of large dimensions so it is necessary to make a dimension reduction. Two processes are performed to achieve a good reduction in size, the first is the selection of movements, proposed methodology of comparison between movements, seeking for which have a greater differentiability and present a lower error in their classification. The second process, is a selection of features, consists of the calculation of contribution of each feature to the classification, is carried out through the WEKA program and its RELIEF algorithm [9, 10]. Finally, the performance comparison of three machine learning techniques, K-Nearest Neighbors (KNN), Artificial Neural Network (ANN) and classifier based on Parzen density. Each of the stages is developed and explained in depth in the text.

The rest of this paper is structured as follows: Section 2 describes the stages of the EMG signal classification procedure for movement identification purposes as well as the database used for experiments. Section 3 presents the proposed

experimental setup. Results, Discussion and Future work are gathered in Section 4.

## 2 Materials and Methods

This section describes the proposed scheme to explore the classification effectiveness on upper limb movements identification through different machine learning techniques. Broadly, our scheme involves stages for preprocessing, segmentation, characterization, movement selection, feature selection and classification as depicted in the explaining block diagram from Fig. 1.

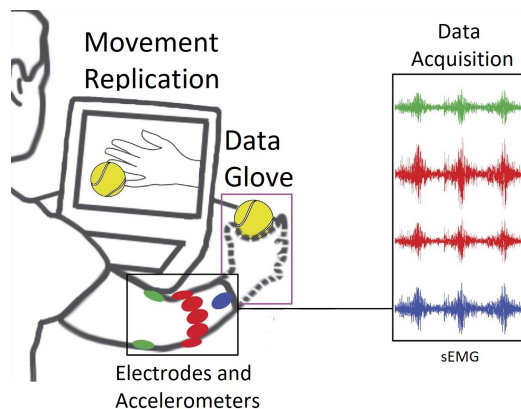


**Fig. 1.** Block diagram of the proposed methodology

### 2.1 Database

The database considered in this study is available at the *Ninaweb* repository from *Ninapro* project [7]. It contains the upper-limb-electromyographic activity of 27 healthy people, performing 52 movements, namely: 12 movements of fingers, 8 isometric and isotonic configurations of the hand, 9 wrist movements and 23 functional grip movements. Such movements were selected from relevant literature as well as rehabilitation guides. Also, the *Ninapro* database includes an acquisition protocol, and brief descriptions of the subjects involved in the data collection. Muscle activity is recorded by 10 double differential electrodes at a sampling rate of 100 Hz. The position of the hand is registered through a *Dataglove* and a inclinometer. The Electrodes are equipped with an amplifier, and a pass filter and a RMS rectifier. The amplification has a factor of 14000. Eight electrodes are placed uniformly around the forearm using an elastic band, at a constant distance, just below the elbow. In turn, two additional electrodes are placed in the long flexor and extensor of the forearm [4].

By obeying experimental protocol, each subject sit down in a chair, in front of a table with a big monitor. While the electrodes, *Dataglove* and inclinometer are working, the patient repeats 10 times each of the 52 movements shown on the screen, as can be seen in Fig. 2. Each repetition takes 5 second, followed by a rest period of 3 seconds.



**Fig. 2.** EMG signal acquisition protocol for Ninapro database oriented to movement identification [4].

## 2.2 Stages of System

*Pre-processing* The amplitude and frequency features of the raw electromyography signal have been shown to be highly variable and sensitive to many factors, like Extrinsic factors (electrode position, skin preparation, among others) and Intrinsic factors (physiological, anatomical and biochemical features of the muscles and others) [3]. Some Normalization procedure is therefore necessary for the conversion of the signal to a scale relative to a known and repeatable value. For the structure of the database, the normalization was applied by electrode. An electrode is taken and the maximum value is found. Then, with this value the whole signal is divided.

*Segmentation* At this stage, a procedure is performed of segmentation. By virtue, the database contains a tags vector, facilitating the trimming of the signals. The vector of tags indicates what action the patient was doing throughout the data collection process. A number 0 means that the patient is at rest and with the rest of the numbers that is making some movement, each movement has an identification number. So the segmentation consists, in taken the signal with its tag vector and eliminate all pauses, leaving only the signals of the movements. The trimmed and normalized electromyographic signals are stored in a data structure.

*Characterization* The matrix of features is organized in the following way: In each row, the data corresponding to patient  $n$  is placed, which is performing a movement  $j$  and a repetition  $k$  and so on. For columns, a bibliographic review is made, obtaining 28 different features for the EMG signals, which are applied to each of the 10 electrodes. As mentioned above, the feature matrix has a size of 14040 per 280. Among the 28 features, there are two types that can be identified. For a better understanding, each one of them is explained below:

- *Temporal features*: Made a reference to the variables that we can get of the signals that find in the time domain, and quantified each T seconds of time. The features used for this study are: Area under the curve, absolute mean value, rms value, standard deviation, variance, median, entropy, energy and power [1, 2, 11].
- *Spectral features*: The time-frequency representation of a signal provides information of the distribution of its energy in the two domains, obtaining a more complete description of the physical phenomenon. The most common techniques used in the extraction of spectral are: the Fourier transform STFT (Short Time Fourier Transform), the continuous wavelet transform CWT, the discrete wavelet transform DWT and the wavelet packet transform WPT [5, 8, 11]. On one hand, the Fourier transform is widely used in the processing and analysis of signals, the results obtained after its application have been satisfactory in cases where the signals were periodic and sufficiently regular. The results are different when analyzing signals whose spectrum varies with time (non-stationary signals). On the other hand, the Wavelet transform is efficient for the local analysis of non-stationary signals and of rapid transience. Like the Fourier Transform with a time window, locates the signal into a time-scale representation. The temporal aspect of the signals is preserved. The difference is that the Transformed Wavelet provides a multi-resolution analysis with dilated window. The transform of a Wavelet function is the decomposition of the function  $f(t)$  into a set of functions  $\psi_{s,\tau}$ , providing a basis called Wavelets  $W_f$ . The wavelet transform is defined as Eq. (1):

$$W_f(s, \tau) = \int f(t)\psi_{s,\tau}(t)dt \quad (1)$$

Wavelets are generated from the transfer and scale change of a function  $\psi(t)$ , called "Mother Wavelet", as detailed in Eq. 2.

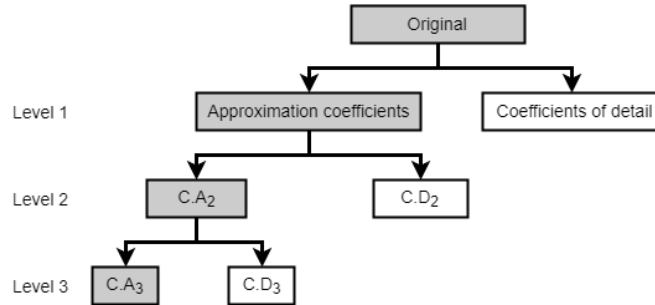
$$\psi_{s,\tau}(t) = \frac{1}{\sqrt{s}}\psi\left(\frac{t-\tau}{s}\right) \quad (2)$$

Where  $s$  is the scale factor, and  $\tau$  is the translation factor.

The wavelet coefficients can be calculated by a discrete algorithm implemented in the recursive application of discrete high-pass and low-pass filters. As shown in Fig. 3.

A wavelet Daubechies 2 (db2) function is used, with 3 levels of decomposition to obtain the coefficients of the filters of analysis and of detail, then features are extracted to the wavelet coefficients as well as for the discretized signal over time.

*Movements Selection* To reduce the number of movements, a comparison methodology is proposed. Involves taking a group of 2 movements, classify them and calculate their average error, then add one more movement and repeat successively the process until finishing with the 52 movements.



**Fig. 3.** Wavelet Transform

Different combinations are made to find the movements that, when classified, have the least possible error. In the event that when adding a movement to the work group, the error goes up abruptly, this movement is eliminated immediately. As a result of this process, we obtained 10 movements, which have a very low classification error and few misclassified objects. It is important to highlight that the classifier used is KNN; each classification was repeated 25 times per group of movements and all functions applied are in the toolbox PRtools.

*Features Selection* The aim of this stage is to decrease the number of variables, deleting redundant or useless information. To improve the classifiers training time, is reduced the computational cost and performance is improved, when carrying out the training with a subset instead of the original data set.

This stage is carried out with the RELIEF algorithm, used in the binary classification, which generalizes to the polynomial classification through different binary problems and giving contribution weights to each feature [9, 10].

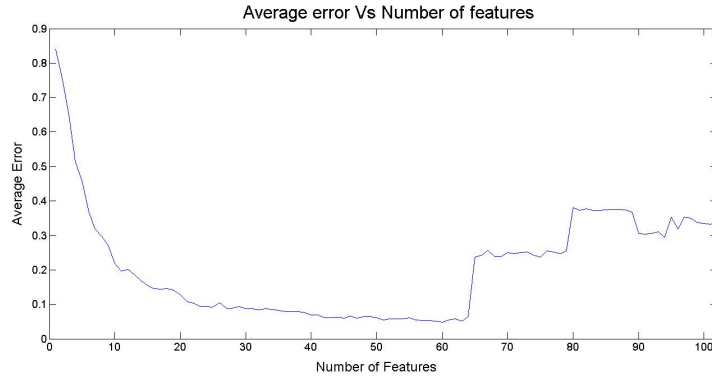
The algorithm orders the features according to their contributions, from highest to lowest, so that the feature matrix is reorganized and those columns or features that do not contribute to the classification of the movements are eliminated.

To decide which number of features are appropriate, tests are performed with the KNN classifier, the number of features are varied each 25 iterations, in each one the movements are classified and the average error is calculated at the end of the iterations. As a result, you get a vector with the average error according to the number of features, as shown in Fig. 4.

Thereby, the number of features is reduced to 60 columns, where the error is minimum. This new feature matrix, is used for the next step, the comparison of each classifiers. As seen In Fig. 5, the block diagram shows the methodology explained previously.

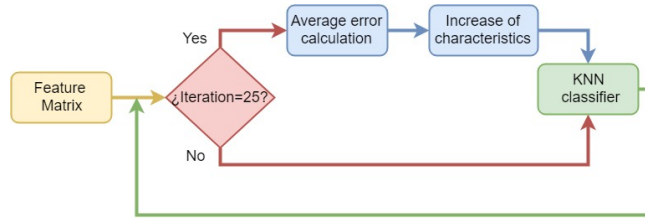
### 2.3 Classification

The final stage of this process is the classification, we carry out a bibliographic review on the algorithms used for the classification of movements with EMG



**Fig. 4.** Results of the selection of features, as the average error varies according to the number of features.

signals. There are many researches and articles on this topic, where different algorithms were used and its performance is good, but we look for an algorithm, which does not take much time in training, its computational cost has been low, and it has been tested in multiple class problems. Continue with the process, with the new feature matrix, the classification of these movements is carried out with the following techniques.



**Fig. 5.** Block diagram of the methodology used to select the optimum number of features

1. K-nearest neighbors (KNN): It is a method of non-parametric supervised classification. A simple configuration is used, it consists in assigning to a sample the most frequent class to which its nearest  $K$  neighbors belong. Having a data matrix that stores  $N$  cases, each of which is defined with  $n$  features  $(X_1...X_n)$ , and a variable  $C$  that defines the class of each sample. The  $N$  cases are denoted:

$$(\mathbf{x}_1, c_1), \dots, (\mathbf{x}_N, c_N), \tag{3}$$

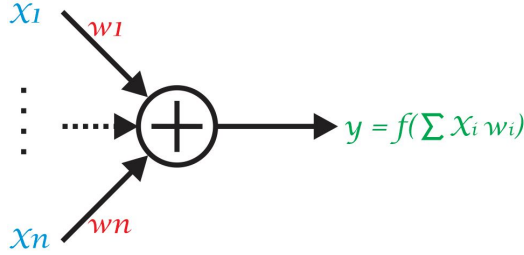


where:

$$\begin{aligned} \mathbf{x}_i &= (x_{i,1} \dots x_{i,n}) \text{ for all } i = 1, \dots, N. \\ c_i &\in \{c^1, \dots, c^m\} \text{ for all } i = 1, \dots, N. \end{aligned}$$

$c_1 \dots c_m$  denote the possible  $m$  values of  $c$ .  $\mathbf{x}$  is the new sample to classify. The algorithm calculates the euclidean distances of the cases classified to the new case  $\mathbf{x}$ . Once the nearest  $K$  cases have been selected,  $\mathbf{x}$  is assigned the most frequent class  $c$ . Empirically through different tests,  $k = 5$  was established [8, 12, 13].

2. Artificial neural network (ANN): Artificial Neuronal Network is heuristic classification technique emulates the behavior of a biological brain through a large number of artificial neurons that connect and they are activated by means of functions. The model of a single neuron can be represented as in Fig. 6. Where  $\mathbf{x}$  denotes the input values or features, each of the  $n$  inputs



**Fig. 6.** Model of a single Neuron

has an associated weight  $w$  (emulates synapsys force). The input values are multiplied by their weights and summed, obtaining:

$$v = w_1 x_1 + w_2 x_2 + \dots w_n x_n = \sum_{i=1}^n w_i x_i. \quad (4)$$

The Neural Network is a collection of neurons connected in a network with three layers: Input layer is associated with the input variables. The Hidden Layer is not connected directly to the environment, but in this layer is where we can calculate each  $w$ . The Output Layer is associated with the output variables and are followed by an activation function. The process of finding a set of weights  $w$  such that for a given input the network produces the desired output is called training [1, 6, 14].

In this work a neural network is trained with a back propagation algorithm with a hidden layer with 10 neurons. The weight initialization consists of setting all weights to be zero, as well as the dataset is used as a tuning set. A sigmoid function is used in this work.

3. Parzen-density-based classifier: Usually the principals classifiers are designed for binary classification, but in practical applications, it is common that the number of classes is greater than two, in our case we have ten different movements to classify [15,16]. So, the Parzen-density-based- classifier is designed to work with multi-class problems. This probabilistic-based classification method requires a smoothing parameter for the Gaussian distribution computation, which is optimized.

### 3 Experimental setup

Importantly for this process, we use the toolbox of Matlab called PRTools, which has all the necessary functions to perform the classification of movements with the different machine learning techniques and the calculation of efficiency of them.

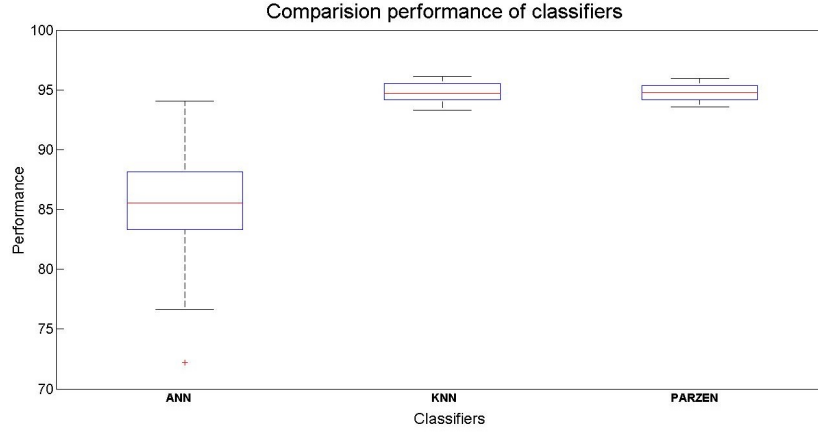
From the new matrix, two groups are obtained in a random way, one of training that uses 75% of the data and the other group of verification with the rest of the data. With the selected groups, we proceed to classify the movements, this step is repeated 30 times with each classifier, and the error and the misclassified movements are stored in a two vectors. At the end, the average error and the deviation are calculated, which are the measures to estimate the effectiveness of classification of the different machine learning techniques used.

### 4 Results, Discussion and Future work

Based on the result of the mean error and the standard deviation, it was clear that in Fig. 7 that the KNN and Parzen-density-based classifier present a better overall performance with 94.82% and 94.8%, while the neural network does not get close to the performance of the other two classifiers, with the 60 selected features reached 85,26% of recognition rate, possibly the number of features or data is not enough for a training of the neural network, since the average number of misclassified movements does not have a big difference with respect to KNN and Parzen, as seen in Table 1. In Fig. 4 shows that it is possible to obtain a good performance with KNN algorithm using only twenty features.

Fig. 7 Also reveals greater uniformity of the KNN and Parzen classifiers in each of the tests, giving a standard deviation of 0.79% for KNN, 0.72% for Parzen-density-based classifier and 4.52% for the Neural Network. These results are comparable to the results obtained in [1], where with back-propagation neural networks a 98.21% performance was obtained but classifying only 5 movements. The recognition rates were 84.9% for the k-NN in [13] where five wrist movements were classified, this article highlight the difficult to place the electrodes in the forearm, but in our case the database used was acquired with a strict protocol avoiding this problem, this makes a difference in the results obtained, Ninapro also has more data acquisition channels, with which it is possible to obtain more information and discern the most important through the RELIEF algorithm.

As future work, it is proposed to develop a comparison with more classifiers such as SVM and LDA. Also observe the result of using variations of the algorithms, such as KNN with weights and FF-ANN. Other parameters of performance as specificity, sensitivity and computational cost must also be evaluated. We will explore the possibility to apply this knowledge in a practical application as a hand prosthesis or human machine interactive in real time using EMG signals of the forearm, searching a high classification rate.



**Fig. 7.** Performance of the classifiers. In the following order, ANN, KNN and Parzen.

**Table 1.** Average of Misclassified movements

Classifiers	Number of movements									
	1	2	3	4	5	6	7	8	9	10
KNN	5	4	4	3	8	3	3	2	2	2
ANN	7	2	3	4	6	4	4	5	2	4
Parzen	5	4	4	3	8	4	2	2	1	2

## 5 Acknowledgments

This work is supported by the Smart Data Analysis Systems - SDAS group (<http://sdas-group.com>), as well as the “Grupo de Investigación en Ingeniería Eléctrica y Electrónica - GIIEE” from Universidad de Nariño. Also, the authors

acknowledge to the research project supported by Agreement No. 095 November 20th, 2014 by VIPRI from Universidad de Nariño.

## References

1. Phinyomark, A., Phukpattaranont, P., Limsakul, C.: A review of control methods for electric power wheelchairs based on electromyography signals with special emphasis on pattern recognition. *IETE Technical Review* **28**(4) (2011) 316–326
2. Aguiar, L.F., Bó, A.P.: Hand gestures recognition using electromyography for bilateral upper limb rehabilitation. In: *Life Sciences Conference (LSC), 2017 IEEE, IEEE* (2017) 63–66
3. Halaki, M., Ginn, K.: Normalization of emg signals: To normalize or not to normalize and what to normalize to? (2012)
4. Atzori, M., Gijsberts, A., Castellini, C., Caputo, B., Hager, A.G.M., Elsig, S., Giatsidis, G., Bassetto, F., Müller, H.: Electromyography data for non-invasive naturally-controlled robotic hand prostheses. *Scientific data* **1** (2014) 140053
5. Podrug, E., Subasi, A.: Surface emg pattern recognition by using dwt feature extraction and svm classifier. In: *The 1st Conference of Medical and Biological Engineering in Bosnia and Herzegovina (CMBEBIH 2015), 13-15 March 2015.* (2015)
6. Vazquez, S.A.V., Oubram, O., Ali, B.: Intelligent recognition system of myoelectric signals of human hand movement. In: *International Symposium on Intelligent Computing Systems, Springer* (2018) 97–112
7. Atzori, M., Gijsberts, A., Kuzborskij, I., Elsig, S., Hager, A.G.M., Deriaz, O., Castellini, C., Müller, H., Caputo, B.: Characterization of a benchmark database for myoelectric movement classification. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* **23**(1) (2015) 73–83
8. Krishna, V.A., Thomas, P.: Classification of emg signals using spectral features extracted from dominant motor unit action potential. *International Journal of Engineering and Advanced Technology* **4**(5) (2015) 196–200
9. Kononenko, I.: Estimating attributes: analysis and extensions of relief. In: *European conference on machine learning, Springer* (1994) 171–182
10. Kira, K., Rendell, L.A.: A practical approach to feature selection. In: *Machine Learning Proceedings 1992.* Elsevier (1992) 249–256
11. Romo, H., Realpe, J., Jojoa, P., Cauca, U.: Surface emg signals analysis and its applications in hand prosthesis control. *Revista Avances en Sistemas e Informática* **4**(1) (2007) 127–136
12. Shin, S., Tafreshi, R., Langari, R.: A performance comparison of hand motion emg classification. In: *Biomedical Engineering (MECBME), 2014 Middle East Conference on, IEEE* (2014) 353–356
13. Kim, K.S., Choi, H.H., Moon, C.S., Mun, C.W.: Comparison of k-nearest neighbor, quadratic discriminant and linear discriminant analysis in classification of electromyogram signals based on the wrist-motion directions. *Current Applied Physics* **11**(3) (2011) 740 – 745
14. Arozi, M., Putri, F.T., Ariyanto, M., Caesarendra, W., Widoyatriatmo, A., Setiawan, J.D., et al.: Electromyography (emg) signal recognition using combined discrete wavelet transform based on artificial neural network (ann). In: *Industrial, Mechanical, Electrical, and Chemical Engineering (ICIMECE), International Conference of, IEEE* (2016) 95–99

15. Pan, Z.W., Xiang, D.H., Xiao, Q.W., Zhou, D.X.: Parzen windows for multi-class classification. *Journal of Complexity* **24**(5) (2008) 606 – 618
16. Kurzynski, M., Wolczowski, A.: Hetero- and homogeneous multiclassifier systems based on competence measure applied to the recognition of hand grasping movements. (2014) 163–174

## **ANEXO B**

### **Artículo publicado en IWAIPR 2018: Progress in Artificial Intelligence and Pattern Recognition**

En la siguiente hoja se presenta el paper sometido al IWAIPR 2018 : 6th Internacional Workshop on Artificial Intelligence and Pattern Recognition, desarrollado en Septiembre de 2018 en La Habana, Cuba.

# Movement identification in EMG signals using machine learning: A comparative study

L. Lasso-Arciniegas<sup>1</sup>, A. Viveros-Melo<sup>1</sup>, J.A. Salazar-Castro<sup>2</sup>, M. A. Becerra<sup>3</sup>,  
A. E. Castro-Ospina<sup>3</sup>, E. J. Revelo-Fuelagán<sup>1</sup>, and D. H. Peluffo-Ordóñez<sup>2,4</sup>

<sup>1</sup> Universidad de Nariño, Colombia,

<sup>2</sup> Corporación Universitaria Autónoma de Nariño, Colombia,

<sup>3</sup> Instituto tecnológico metropolitano, Colombia,

<sup>4</sup> Yachay Tech, Urcuqui, Ecuador

**Abstract.** The analysis of electromyographic (EMG) signals enables the development of important technologies for industry and medical environments, due mainly to the design of EMG-based human-computer interfaces. There exists a wide range of applications encompassing: Wireless-computer controlling, rehabilitation, wheelchair guiding, and among others. The semantic interpretation of EMG analysis is typically conducted by machine learning algorithms, and mainly involves stages for signal characterization and classification. This work presents a methodology for comparing a set of state-of-the-art approaches of EMG signal characterization and classification within a movement identification framework. We compare the performance of three classifiers (KNN, Parzen-density-based classifier and ANN) using spectral (Wavelets) and time-domain-based (statistical and morphological descriptors) features. Also, a methodology for movement selection is proposed. Results are comparable with those reported in literature, reaching classification performance of  $(90.89 \pm 1.12)\%$  (KNN),  $(93.92 \pm 0.34)\%$  (ANN) and  $91.09 \pm 0.93$  (Parzen-density-based classifier) with 12 movements.

**Keywords:** ANN, EMG signals, Feature extraction, KNN, Parzen.

## 1 Introduction

Nowadays, technologies for human-computer interfaces (HCIs) are being widely used, as they are becoming indispensable for people life's activities. Thanks to the development of fields such as pattern identification, signal processing, among others, there have been proposed different tools and systems to control a machine through the use of signals, as it is the case of Electromyographic (EMG) signals. Some outstanding applications of EMG-based HCI are: Wireless-computer controlling through hand movement, wheelchair directing/guiding with finger motions, and rehabilitation [1, 2].

Once acquired and digitalized, and since being electrical-unidimensional-type signals, the ECG signals are analyzed within a process mostly involving stages for preprocessing, characterization, data representation, and classification [3]:

Firstly, preprocessing is aimed at filtering (removing artifacts and noisy information) as well as adjusting the signal for subsequent tasks (some processing procedures are sensitive to length or amplitude). Secondly, a characterization task is performed for extracting descriptive features or attributes (especially is required when using directly the amplitude values is unfeasible for length, informative, or computational cost reasons). Thirdly, to select or best represent the afore obtained features in terms of separability, a data representation may be used. Finally, the automatic classification process is applied over the data to assign to relate the EMG signal with an instruction (for instance, a movement in a HCI-based device). All these stages have addressed for some studies [4–6], but a highly accurate system for EMG semantic interpretation is still a problem to be solved.

Based on state-of-the-art techniques, this work proposes a methodology for EMG signal characterization and classification techniques to identify movements. In particular, the spectral features (wavelet coefficients, average frequency, peak frequency, Kurtosis, Shannon, Fischer asymmetry coefficient), temporal features and statistics are used (Area under the curve, absolute mean value, effective value, standard deviation, variance, median, Willison amplitude, Histogram, simple square integral entropy) [1, 4, 7–12]. The characterization of the signal, leaves a matrix of large dimensions so it is necessary to make a dimension reduction. Two processes are performed to achieve a good reduction in size, the first is the selection of movements, proposed methodology of comparison between movements, seeking for which have a greater differentiability and present a lower error in their classification. The second process, is a selection of features, consists of the calculation of contribution of each feature to the classification, is carried out through the WEKA program and its RELIEF algorithm [13, 14]. Finally, the performance comparison of three machine learning techniques, k-nearest neighbors (KNN), Artificial neural network (ANN) and classifier based on Parzen density. Each of the stages is developed and explained in depth in the text.

The rest of this paper is structured as follows: Section 2 describes the stages of the EMG signal classification procedure for movement identification purposes as well as the database used for experiments. Section 3 presents the proposed experimental setup. Results and discussion are gathered in Section 4. Finally, Conclusions and future work in Section 5

## 2 Materials and Methods

This section describes the methodology used to carry out the comparative study and identify a group of features that allows get a high performance of classifiers. In general, our research contains the following stages: preprocessing, segmentation, characterization, dimension reduction and finally classifiers as shown in the explaining block diagram from Fig. 1.



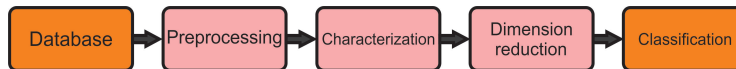


Fig. 1. Block diagram of the proposed methodology

2.1 Database

For this research, we considered the database that is available *Ninaweb* repository from *Ninapro* project [7]. The *Ninapro* project describes the acquisition protocol and the subjects involved in the data acquisition. The protocol said that muscle activity is recorded by 10 double differential electrodes, 8 electrodes are located uniformly around the forearm using an elastic band, at a constant distance, just below the elbow. In turn, two additional electrodes are placed in the long flexor and extensor of the forearm. The hand position is taken by a *Data-glove* and an inclinometer. The sample rate used is for 100 Hz and amplification factor is for 14000.

According to experimental protocol, each subject sit down in front of a screen where it shown the 52 movements, the patient repeats 10 times each one of the movements. All movements are recorded by the electrodes and *Dataglove*, as can be seen in Fig. 2. Each repetition takes 5 second, followed by a rest period of 3 seconds.

The *Ninapro* database include the electromyographic activity of upper-limb of 27 healthy people executing 52 movements, which are divided as: 12 movements of fingers, 8 isometric and isotonic configurations of the hand, 9 wrist movements and 23 functional grip movements [4].

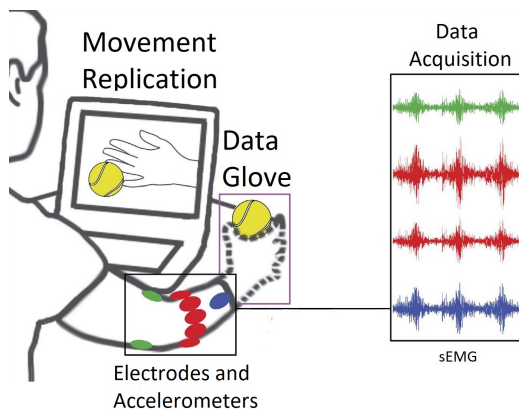


Fig. 2. EMG signal acquisition protocol for *Ninapro* database oriented to movement identification [4].

## 2.2 Stages of System

*Pre-processing* EMG signals can be affected by several factors, such as skin preparation, anatomical and biochemical features of the muscles and among others [15]. For the above, amplitude and frequency features are also affected, so it is necessary to compensate the effect of these factors with some procedure, in this research we applied a normalization and segmentation procedure. For the form of the database, the normalization was applied to each electrode, we took an electrode and found the maximum value of the signal of this electrode, finally the whole signal is divided with this maximum value. The Ninapro database contains a tag vector, which we used for segmentation procedure. The tag vector lets identify when the patient was doing a movement, in addition what kind of movement. So this vector facilitates the trimming of the signals.

*Characterization* The matrix of features is organized in the following way: In each row, the data corresponding to patient  $n$  is placed, which is performing a movement  $w$  and a repetition  $k$  and so on. For columns, a bibliographic review is made, obtaining 45 different features for the EMG signals, which are applied to each of the 10 electrodes. As mentioned above, the feature matrix has a size of 14040 per 450. This matrix is organized as shown forward in the next matrix.

$$X = [x_{i,j}] = \begin{pmatrix} x_{1,1} & x_{1,2} & \dots & x_{1,d} \\ x_{2,1} & x_{2,2} & \dots & x_{2,d} \\ \vdots & \vdots & \ddots & \vdots \\ x_{M,1} & x_{M,2} & \dots & x_{M,d} \end{pmatrix}$$

where  $X \in \mathbb{R}^{M \times d}$  is the matrix of features,  $M$  is the number of samples = 14040 corresponding to 270 repetitions  $k \times 52$  classes  $w$ .  $d$  is the total number of features = 450 corresponding to 45 attributes  $\times$  10 electrodes.

Among the 45 features, there are two types that can be identified, temporal and spectral features. For a better understanding, each one of them is explained below.

- *Temporal features*: Makes a reference to the variables that we can get of the signals that find in the time domain, and quantified each  $T$  seconds of time. The features used are explain below.
  1. Area under the curve: In general, we use like a index for detected the begins of muscular activity. It feature have a relation with activation point of signal [9, 10, 12].

$$IEMG = \sum_{n=1}^N |x_n| \quad (1)$$

2. Mean absolute value: Estimate the absolute mean value  $\overline{X}_i$ , on each segment  $i$  of  $N$  samples of the signal [1,16], through expression:

$$\overline{X}_i = \frac{1}{N} \sum_{n=1}^N |x_n|, i = 1, 2, 3 \dots, I \quad (2)$$

3. Root mean square value (RMS): [9].

$$RMS = \sqrt{\frac{1}{N} \sum_{n=1}^N x_n^2} \quad (3)$$

4. Variance of EMG (VAR): [1,10].

$$VAR = \frac{1}{N-1} \sum_{n=1}^N x_n^2 \quad (4)$$

5. Entropy: Entropy is the statistical measure of randomness that can be used to characterize.

$$EP = \sum P * \log_2 P \quad (5)$$

Suppose that a random variable has a  $k$  possible states and suppose all states are equally probable. Then the probability of it becoming one of those combinations will be  $P = \frac{1}{k}$ .

6. Simple square integral (SSI): [9,10]

$$SSI = \sum_{n=1}^N |x_n|^2 \quad (6)$$

7. Histogram: Divide the elements into EMG signal in  $b$  equis spaced segments and return the number of elements in each segment. EMG is an extension version of the WAMP feature [10].
8. Willison Amplitude (WAMP): It is the number of times that the difference between the amplitude of the EMG signal of two adjacent segments exceeds a predefined threshold to reduce the effects of noise [9,10].

$$WAMP = \sum_{n=1}^{N-1} f(x_n - x_{n+1}) \quad (7)$$

$$f(x) = \begin{cases} 1 & \text{If } x \geq \text{threshold} \\ 0 & \text{otherway} \end{cases}$$

9. Waveform length: It is the cumulative length of the waveform over the time segment. WL is related to the amplitude, frequency and time of the waveform [1,9,10,16]. It is given by:

$$WL = \sum_{n=1}^{N-1} |x_{n+1} - x_n| \quad (8)$$

- *Spectral features*: The time-frequency representation of a signal provides information of the distribution of its energy in the two domains, obtaining a more complete description of the physical phenomenon. The most common techniques used in the extraction of spectral are:

1. *Curtosis*: It is used to describe the distribution and a feature that identifies the trend of the maximum data. The Curtosis data level is determined by comparing the peak of the inclination curve of the data distribution and the normal curve.

$$Curtosis = \frac{\sum_{n=1}^{N-1} (x_i - \bar{x})^4}{Ns^4} - 3 \quad (9)$$

Where  $\bar{x}$  is the mean and  $s$  the standard deviation.

2. *Shanon*: [11]

$$E_s = -\frac{1}{M} \sum_k c(k)^2 \log(c(k)^2) \quad (10)$$

3. *Fischer asymmetry coefficient*: [17]

$$CA_f = \frac{\sum_{i=1}^N (x_i - \bar{x}^3)}{Ns^3} \quad (11)$$

Where  $\bar{x}$  is the mean and  $s$  the standard deviation.

4. *Peak frequency*: The maximum frequency is the frequency at which the maximum power is produced [8].

$$PF = \max(P_i), i = 1, 2, 3 \dots, n \quad (12)$$

5. *Average frequency*: [8]

$$AF = \frac{\sum_{i=1}^N P_i}{n} \quad (13)$$

6. *Modified Average Frequency*: It is calculated as the sum of the product of the amplitude spectrum and the frequency, divided by the sum total of the intensity of the spectrum [10]

$$MMNF = \frac{\sum_{j=1}^M f_j A_j}{\sum_{j=1}^M A_j} \quad (14)$$

Where  $f_j$  is the frequency of the spectrum.

7. *Wavelet transform*:

The Wavelet transform is efficient for the local analysis of non-stationary signals and of rapid transience. Like the Fourier Transform with a time window, locates the signal into a time-scale representation. The temporal aspect of the signals is preserved. The difference is that the Transformed

Wavelet provides a multi-resolution analysis with dilated window [5, 8, 16]. The transform of a Wavelet function is the decomposition of the function into a set of functions, providing a basis called Wavelets, defined as Eq. (15):

$$W_f(s, \tau) = \int f(t)\psi_{s,\tau}(t)dt \quad (15)$$

Wavelets are generated from the transfer and scale change of a function called Mother Wavelet, as detailed in Eq. 16.

$$\psi_{s,\tau}(t) = \frac{1}{\sqrt{s}}\psi\left(\frac{t-\tau}{s}\right) \quad (16)$$

The wavelet coefficients can be calculated by a discrete algorithm implemented in the recursive application of discrete high-pass and low-pass filters. A wavelet Daubechies 2 (db2) function is used, with 3 levels of decomposition to obtain the coefficients of the filters of analysis and detail, then features are extracted to the wavelet coefficients as well as for the discretized signal over time. To each wavelet decomposition level, we applied temporal and spectral features.

*Movements Selection* In human-machine control interfaces with EMG signals, the number of movements that the machine be able to classify correctly, will be the degrees of freedom to control process. Many movements can difficult the classification problem, due to this its necessary to identify the greater number of classes that it can classify, with an appropriate performance. The database has 52 classes  $w$ , in virtue of the above problem mention, we suggested a criterion based on euclidean distance (ED) and standard deviation (SD).

From the characteristics matrix described in section 2.2 its calculated the centroid  $\mathbf{c}$  of the points cloud (PC) for each movement, as the mean value of the repetitions  $k$ , belonging to each  $w^{th}$  class. Euclidean Distance of each class its calculated as the root of square differences between co-ordinates of a pair of centroids. The ED, is defined as:

$$ED(\mathbf{c1}, \mathbf{c2}) = \sqrt{(c1_1 - c2_1)^2 + c1_2 - x2_2)^2 + \dots + (c1_n - c2_n)^2}, \quad (17)$$

where  $c1_n$  and  $c2_n$  are the  $n^{th}$  components of centroids  $\mathbf{c}$ . Also standard deviation is calculated of the points cloud of each class. SD is calculated as:

$$SD = \sqrt{\frac{\sum(x_{k_n,d_n} - \mu)^2}{N}}, \quad (18)$$

where  $x_{(k_n,d_n)}$  is the data corresponding to the characteristic  $d_n$  of the repetition  $k$ , and  $\mu$  is the mean value. Euclidean distance between centroids allows evaluate how much far away are the classes, and the standard deviation says what so compact is a points cloud. To improve the performance of the classify its necessary that PC of each movement be far away and as compacts as

possible. So a parameter that allows to evaluate these two features between two different classes is defined as:

$$P(w_1, w_2) = \frac{ED(\mathbf{c1}, \mathbf{c2})}{SD_1 + SD_2}, \quad (19)$$

where  $ED(\mathbf{c1}, \mathbf{c2})$  is the euclidean distance between the centroids of the class 1 ( $w_1$ ) and class 2 ( $w_2$ ),  $SD_1$  and  $SD_2$  are the standard deviation of classes 1 and 2. SD must be inversely proportional to parameter in order to maximize  $P$  and improve the classification process. This equation is applied to all class combinations. The movements for which the parameter  $P$  is greater are selected. This methodology showed that with the characteristics described in section 2, the movements that present the best separability are: contraction and extension of middle and ring fingers, wrist contraction and extension, writing tripod grasp, tip pinch grasp, lateral grasp, power disc grasp and turn a screw. Several of these movements have great activity of the flexor digitorum superficialis and of the extensor digitorum superficialis.

*Features Selection* In the feature selection process a subset of the original data set is taken. To obtain this subset the Relief algorithm is used. Relief calculates a feature weight for each feature which can then be applied to rank and select top scoring features. RELIEF detects the interactions between attributes to determine those that are most relevant and those that work well in groups. The iterative algorithm consists of randomly selecting an X data, and then select the nearest sample of the same class (Hit) and the nearest one of another class (Miss). Increase the weight of those attributes that have the same value for the Hit instance and different value for the instance Miss.

$$W_i = W_i - (x_i - Hit)^2 + (x_i - Miss)^2 \quad (20)$$

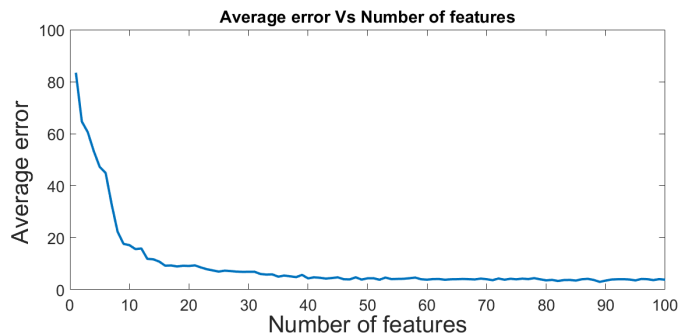
Thus the weight of any given feature decreases if it differs from that feature in nearby instances of the same class more than nearby instances of the other class, and increases in the reverse case.

To select a suitable number of characteristics to train the classifiers, a rapid test is carried out with the KNN classifier. Evaluating its error performance by varying the number of characteristics. From this test Fig. 3. is obtained. As you can see in Fig. 3, 20 characteristics are enough to obtain a good performance. This 20 features are used for the next step, the comparison of each classifiers.

### 2.3 Classification

With the new feature matrix, the classification of these movements is carried out with the following techniques [1, 6, 8, 18, 19].

1. K-nearest neighbors (KNN): It is a method of non-parametric supervised classification, which utilizes a distance measure relative to the k closest neighbors of a point to assign a class to a given data record. The number of the nearest neighbors is optimized with respect to the leave-one-out error on training set.



**Fig. 3.** Results of the selection of features, as the average error varies according to the number of features.

2. Artificial neural network (ANN): This heuristic classification technique emulates the behavior of a biological brain through a large number of artificial neurons that connect and they are activated by means of functions. The neurons are distributed in hidden layers, in this work a neural network is trained with a back-propagation algorithm with a hidden layer with 10 neurons. The weight initialization consists of setting all weights to be zero. Sigmoidal functions are used in the output. The dataset is used for training, validation and test.
3. Parzen-density-based classifier: This probabilistic-based classification method requires a smoothing parameter for the Gaussian distribution computation, which is optimized.

### 3 Experimental setup

To carry out the different experiments of this stage, we used the toolbox of Matlab called PrTools. This toolbox has all necessary functions as classifiers and performance indicators. From the reduced feature matrix, we obtain two sets in a random way, each group is balanced in the number of samples. Classifiers are trained with 75% of the data and validated with the remaining 25%.

To ensure the repeatability of the experiment, and that the results do not depend on a specific data group, the tests are repeated 30 times with the three classifiers. In each test the percentage of performance is obtained and at the end of the 30 repetitions an average of the tests is obtained. The results of each test are reported in the next section.

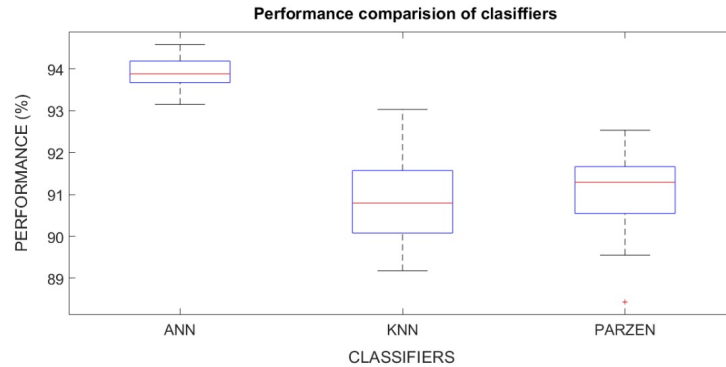
### 4 Results and Discussion

Based on the tests carried out and the performance results recorded in Fig. 4 it is clear that the ANN present a better performance than KNN and Parzen.

Nevertheless all classifiers have error percentage under 10%. The performance of ANN is 93.92%, KNN is 90.89% and PARZEN is 91.09% using 20 features.

Fig. 4 also reveals a greater uniformity of the ANN than KNN and Parzen classifiers in each of the tests, giving a standard deviation of 1.12% for KNN, 0.93% for Parzen-density-based classifier and 0.35% for the neural network. This results are comparable to [1], where with back-propagation neural networks a 98.21% performance was obtained but classifying only 5 movements what limits the control options. In [18] the results of pattern recognition EMG signal using wavelet transform and Artificial Neural Network (ANN) classification has an accuracy rate of 77.5% identifying 8 movements. For this reason it is emphasized that an adequate search of EMG signal characteristics in combination with the appropriate selection of movements allows obtaining a good performance of the ANN classifier. The recognition rates were 84,9% for the KNN in [20] where five wrist movements were classified. It is considerable that this result is due to the appropriate selection of movements based on Euclidean Distance criteria, also used by the KNN classifier.

The process of selection of attributes showed that the most relevant features were: Maximum concentration of histogram, standard deviation, Willison amplitude, modified average frequency, entropy of the signal in the time domain and entropy of the first level of decomposition of the wavelet transform. Most of the relevant information is concentrated in 4 electrodes, corresponding to the flexor and extensor muscles of the forearm.



**Fig. 4.** Performance of the classifiers. In the following order, ANN, KNN and Parzen.

## 5 Conclusions and Future work

This work presents a evaluation of different features of signal analysis, a dimension reduction method and a criterion to evaluate separability of classes. With the aim to improve movements classification with EMG signals.



Experimentally we prove that with the selected movements and the features used it is possible to implement a human - machine control interface with good performance. On the other hand a set of features that help in the movement recognition of upper limb is proposed. We identify 12 movements that with the characteristics described in section 2 present a greater separability, it was also possible to identify the muscles that have the most intervention on these movements.

As a future work, we will explore the possibility to apply this knowledge in a prosthesis prototype with own signals of EMG.

## 6 Acknowledgments

This work is supported by the Smart Data Analysis Systems - SDAS group (<http://sdas-group.com>), as well as the “Grupo de Investigación en Ingeniería Eléctrica y Electrónica - GIIEE” from Universidad de Nariño. Also, the authors acknowledge to the research project supported by Agreement No. 095 November 20th, 2014 by VIPRI from Universidad de Nariño.

## References

1. Phinyomark, A., Phukpattaranont, P., Limsakul, C.: A review of control methods for electric power wheelchairs based on electromyography signals with special emphasis on pattern recognition. *IETE Technical Review* **28**(4) (2011) 316–326
2. Aguiar, L.F., Bó, A.P.: Hand gestures recognition using electromyography for bilateral upper limb rehabilitation. In: *Life Sciences Conference (LSC), 2017 IEEE, IEEE (2017)* 63–66
3. Rodríguez-Sotelo, J., Peluffo-Ordóez, D., Cuesta-Frau, D., Castellanos-Domínguez, G.: Unsupervised feature relevance analysis applied to improve ecg heartbeat clustering. *Computer Methods and Programs in Biomedicine* **108**(1) (2012) 250 – 261
4. Atzori, M., Gijsberts, A., Castellini, C., Caputo, B., Hager, A.G.M., Elsig, S., Giatsidis, G., Bassetto, F., Müller, H.: Electromyography data for non-invasive naturally-controlled robotic hand prostheses. *Scientific data* **1** (2014) 140053
5. Podrug, E., Subasi, A.: Surface emg pattern recognition by using dwt feature extraction and svm classifier. In: *The 1st Conference of Medical and Biological Engineering in Bosnia and Herzegovina (CMBEBIH 2015), 13-15 March 2015.* (2015)
6. Vazquez, S.A.V., Oubram, O., Ali, B.: Intelligent recognition system of myoelectric signals of human hand movement. In: *International Symposium on Intelligent Computing Systems, Springer (2018)* 97–112
7. Atzori, M., Gijsberts, A., Kuzborskij, I., Elsig, S., Hager, A.G.M., Deriaz, O., Castellini, C., Müller, H., Caputo, B.: Characterization of a benchmark database for myoelectric movement classification. *IEEE Transactions on Neural Systems and Rehabilitation Engineering* **23**(1) (2015) 73–83
8. Krishna, V.A., Thomas, P.: Classification of emg signals using spectral features extracted from dominant motor unit action potential. *International Journal of Engineering and Advanced Technology* **4**(5) (2015) 196–200

9. Negi, S., Kumar, Y., Mishra, V.: Feature extraction and classification for emg signals using linear discriminant analysis. In: Advances in Computing, Communication, & Automation (ICACCA)(Fall), International Conference on, IEEE (2016) 1–6
10. Phinyomark, A., Limsakul, C., Phukpattaranont, P.: A novel feature extraction for robust EMG pattern recognition. CoRR **abs/0912.3973** (2009)
11. Ahlstrom, C., Hult, P., Rask, P., Karlsson, J.E., Nylander, E., Dahlström, U., Ask, P.: Feature extraction for systolic heart murmur classification. *Annals of biomedical engineering* **34**(11) (2006) 1666–1677
12. Han, J.S., Song, W.K., Kim, J.S., Bang, W.C., Lee, H., Bien, Z.: New emg pattern recognition based on soft computing techniques and its application to control of a rehabilitation robotic arm. In: Proc. of 6th International Conference on Soft Computing (IIZUKA2000). (2000) 890–897
13. Kononenko, I.: Estimating attributes: analysis and extensions of relief. In: European conference on machine learning, Springer (1994) 171–182
14. Kira, K., Rendell, L.A.: A practical approach to feature selection. In: Machine Learning Proceedings 1992. Elsevier (1992) 249–256
15. Halaki, M., Ginn, K.: Normalization of emg signals: To normalize or not to normalize and what to normalize to. (2012)
16. Romo, H., Realpe, J., Jojoa, P., Cauca, U.: Surface emg signals analysis and its applications in hand prosthesis control. *Revista Avances en Sistemas e Informática* **4**(1) (2007) 127–136
17. López, E.O., Oubram, O., Aguilar, J.V., Vázquez, S.V.: Sistema de reconocimiento inteligente de señales mioeléctricas del movimiento de mano humana. *Revista Académica de la Facultad de Ingeniería* **21**(2) (2017) 41 – 53
18. Arozi, M., Putri, F.T., Ariyanto, M., Caesarendra, W., Widyotriatmo, A., Setiawan, J.D., et al.: Electromyography (emg) signal recognition using combined discrete wavelet transform based on artificial neural network (ann). In: Industrial, Mechanical, Electrical, and Chemical Engineering (ICIMECE), International Conference of, IEEE (2016) 95–99
19. Shin, S., Tafreshi, R., Langari, R.: A performance comparison of hand motion emg classification. In: Biomedical Engineering (MECBME), 2014 Middle East Conference on, IEEE (2014) 353–356
20. Kim, K.S., Choi, H.H., Moon, C.S., Mun, C.W.: Comparison of k-nearest neighbor, quadratic discriminant and linear discriminant analysis in classification of electromyogram signals based on the wrist-motion directions. *Current Applied Physics* **11**(3) (2011) 740 – 745

## **ANEXO C**

### **CERTIFICADO CONGRESO 13 CCC**

Se presenta el certificado como ponentes en el 13vo Congreso Colombiano de Computación (13CCC), desarrollado en Septiembre de 2018 en Cartagena de Indias, Colombia. Link del articulo



Universidad  
Tecnológica  
de Bolívar

CARTAGENA DE INDIAS

## LA UNIVERSIDAD TECNOLÓGICA DE BOLÍVAR

Confiere el presente certificado de ponentes a

Brayan Andres Viveros Melo, Laura Daniela Lasso Arciniegas,  
Jose Alejandro Salazar Castro, Diego Hernán Peluffo Ordoñez,  
Miguel A. Becerra, Andrés Eduardo Castro Ospina,  
Edgardo Javier Revelo Fuelagán.

Con el trabajo:

Exploration of characterization and classification techniques for movement  
identification from EMG signals: Preliminary results

Por asistir al:




**CONGRESO  
COLOMBIANO  
DE COMPUTACIÓN  
2018**



realizado el 26, 27 y 28 de septiembre de 2018

Cartagena de Indias



  
Jairo Enrique Serrano Castañeda  
Organizador 13CCC

  
Juan Carlos Martínez Santos

Organizador 13CCC

## ANEXO D

### Página web EMG.

Se muestra una página web creada con el fin de compartir el desarrollo, vídeo tutorial y artículos que resultaron de este proyecto.

La página se puede encontrar en: <https://sites.google.com/site/degreethesisdiegopeluffo/emg-rehabilitation>

Figura 47. Página web "Classification for movement identification from EMG signals".

**Classification for movement identification from EMG signals**

Laura Daniela Lasso Arciniegas and Brayan Andres Viveros Meo, Universidad de Narino, San Juan de Pasto, Colombia-2019

Electromyographic signals are electrical impulses produced by the muscles during the processes of contraction and relaxation, this feature allows analyzing the physiological state of the muscles and also allows creating control interfaces for communication between user and machine. Among the different applications of interfaces, those dedicated to individuals with motor disabilities that impede the performance of daily activities have taken off. Therefore it is pertinent to perform a search of characteristics and evaluation of machine learning techniques, to define criteria that allow carrying out this type of control interfaces.

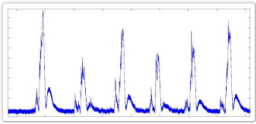


Figure 1. Example of an electromyographic signal.

In order to carry out a classification task, it is necessary to carry out an adequate segmentation, characterization, dimension reduction and finally classification procedure. Finally, once a motor intention order is classified, it must be executed by an actuator of interest.

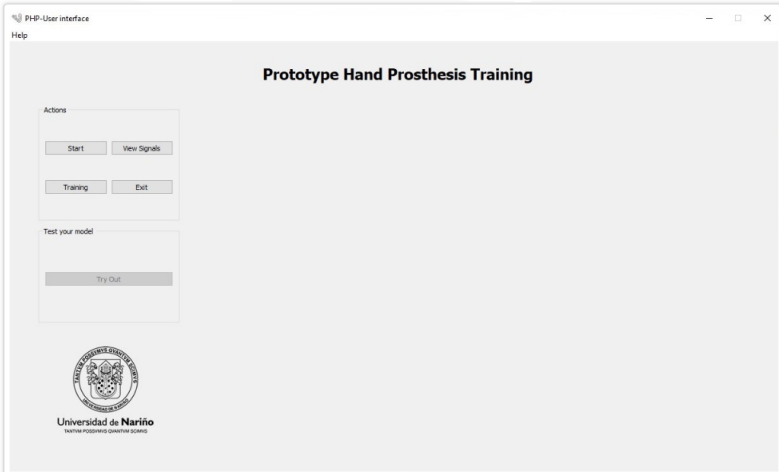
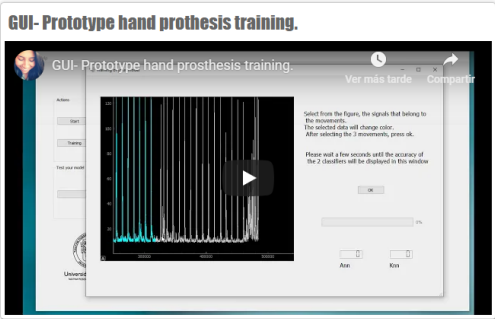


Figure 2. View of the interface implemented on python supported with PyQt Designer.



**Publications.**

**Thesis.**

Exploration of Characterization and Classification Techniques for Movement Identification from EMG Signals: Preliminary Results. Colombian Conference on Computing 2018.  
 A. Viveros-MeloL, Lasso-Arciniegas, J. A. Salazar-CastroD, H. Peluffo-OrdóñezM, A. BecerraA, E. Castro-OspinaE, J. Revelo-Fuelagán.  
[See full paper](#)

Movement Identification in EMG Signals Using Machine Learning: A Comparative Study  
 International Workshop on Artificial Intelligence and Pattern Recognition 2019  
 Laura Lasso-Arciniegas, Andres Viveros-MeloJosé A. Salazar-CastroMiguel A. BecerraAndrés Eduardo Castro-OspinaE, Javier Revelo-FuelagánDiego H. Peluffo-Ordóñez  
[See full paper](#)

**About the authors.**

Andres Viveros Electronic Engineer brayanandres123@hotmail.es	Laura Lasso Electronic Engineer laurad.lasso.a@gmail.com
---	--