



REPÚBLICA DEL ECUADOR

Escuela Politécnica Nacional

" E SCIENTIA HOMINIS SALUS "

La versión digital de esta tesis está protegida por la Ley de Derechos de Autor del Ecuador.

Los derechos de autor han sido entregados a la "ESCUELA POLITÉCNICA NACIONAL" bajo el libre consentimiento del (los) autor(es).

Al consultar esta tesis deberá acatar con las disposiciones de la Ley y las siguientes condiciones de uso:

- Cualquier uso que haga de estos documentos o imágenes deben ser sólo para efectos de investigación o estudio académico, y usted no puede ponerlos a disposición de otra persona.
- Usted deberá reconocer el derecho del autor a ser identificado y citado como el autor de esta tesis.
- No se podrá obtener ningún beneficio comercial y las obras derivadas tienen que estar bajo los mismos términos de licencia que el trabajo original.

El Libre Acceso a la información, promueve el reconocimiento de la originalidad de las ideas de los demás, respetando las normas de presentación y de citación de autores con el fin de no incurrir en actos ilegítimos de copiar y hacer pasar como propias las creaciones de terceras personas.

Respeto hacia sí mismo y hacia los demás.

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

IMPLEMENTACIÓN DE UN SISTEMA DE CLASIFICACIÓN DE GESTOS DEL BRAZO HUMANO UTILIZANDO MYO ARMBAND PARA MANDO A DISTANCIA DE UN BRAZO ROBÓTICO DE 3GDL

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE
“INGENIERO EN ELECTRÓNICA Y CONTROL”**

JONATHAN ALEJANDRO ZEA GUACHAMÍN

DIRECTOR: MARCO ESTEBAN BENALCÁZAR PALACIOS, PhD.

Quito, diciembre 2017

AVAL

Certifico que el presente trabajo fue desarrollado por el señor Jonathan Alejandro Zea Guachamín, bajo mi supervisión.

MARCO E. BENALCÁZAR PH. D.
DIRECTOR DEL TRABAJO DE TITULACIÓN

DECLARACIÓN DE AUTORÍA

Yo, Jonathan Alejandro Zea Guachamín, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Jonathan Alejandro Zea Guachamín

DEDICATORIA

Dedico este trabajo a mis padres, a la So y al Gino; mi familia, que por mucho tiempo esperó para verme como ingeniero.

Jona,

AGRADECIMIENTOS

Ningún buen trabajo se puede realizar sin ayuda alguna. En primer lugar, agradezco a mi papá, Roberto Zea, la primera persona que vio mi potencial, y que es la que más cree en mí de todo el mundo.

A mi mamá, Fanny Guachamín, la más inteligente y empeñosa de la casa, la única que se da cuenta, y la que siempre presume orgullosa cada uno de mis pequeños logros.

Al Gino, por ser mi hermano el único que me ha acompañado casi toda mi vida.

A la Sofía, a quien quiero a pesar de ser la hermana gemela malvada.

A los chicos CEDIA, donde sea que estén; a la Gata, y a todos mis amigos en todas partes.

A Diana Álvarez, que me obliga a seguir porque todavía está muy lejos.

A los profesionales que conocí en la universidad como el doctor Marco Benalcázar, que me mostró las responsabilidades y obligaciones propias de un verdadero profesional y ser humano.

A Herb y Tania de la OPSI, que luchan todos los días en silencio para que más estudiantes salgan y conozcan el mundo.

Y finalmente, agradezco a la Escuela Politécnica Nacional que, a pesar de todo, sigue siendo La Politécnica por antonomasia.

Jonathan Alejandro Zea G.

ÍNDICE DE CONTENIDO

AVAL.....	I
DECLARACIÓN DE AUTORÍA	II
DEDICATORIA	III
AGRADECIMIENTOS	IV
ÍNDICE DE CONTENIDO.....	V
RESUMEN	VIII
ABSTRACT.....	IX
1. INTRODUCCIÓN.....	1
1.1 Objetivos.....	2
1.2 Alcances	2
1.3 Marco Teórico	3
Descripción de las señales electromiográficas.....	3
<i>Señal electromiográfica</i>	<i>4</i>
<i>Adquisición de la señal electromiográfica.....</i>	<i>5</i>
<i>Modelo matemático de la señal electromiográfica basado en la parte biológica.....</i>	<i>6</i>
<i>Modelo matemático de la señal electromiográfica basado en procesos gaussianos</i>	<i>8</i>
Myo Armband	9
<i>Características del Myo Armband</i>	<i>9</i>
Sistemas de clasificación de gestos.....	10
<i>Etapas de los sistemas de clasificación de gestos.....</i>	<i>10</i>
<i>Adquisición de señales.....</i>	<i>11</i>
<i>Extracción de características</i>	<i>12</i>
<i>Clasificadores</i>	<i>13</i>
Aplicaciones de los sistemas de clasificación de gestos	15
<i>Sistemas de clasificación de gestos aplicados a la robótica.....</i>	<i>15</i>
2. METODOLOGÍA.....	16
2.1 Adquisición de datos del Myo Armband	16
Librería MyoMex	17
Colocación del Myo Armband	18
Representación de la Señal EMG medida por el Myo Armband.....	18
2.2 Entrenamiento del sistema de clasificación de gestos.....	19
Definición de los gestos para la clasificación	20

Señales Electromiográficas de los gestos de entrenamiento	21
2.3 Clasificación de gestos en tiempo real	23
Preprocesamiento de la señal electromiográfica	24
<i>Descripción del Filtro de Butterworth</i>	25
<i>Rectificación y filtrado de las señales EMG</i>	26
Comparación de las señales electromiográficas utilizando DTW	29
<i>Descripción del algoritmo de alineamiento temporal dinámico (DTW)</i>	29
<i>Cálculo de la distancia DTW entre las señales EMG</i>	35
Clasificación de las señales electromiográficas utilizando kNN	36
<i>Descripción del clasificador de los k vecinos más cercanos (kNN)</i>	36
<i>Clasificación utilizando kNN</i>	38
Etapa de posfiltrado de la señal	40
2.4 Mando a distancia del brazo robótico de 3 GDL	42
Configuración del brazo robótico	42
<i>Kit Lego Mindstorms EV3</i>	43
<i>Implementación del brazo robótico de 3 GDL</i>	45
Codificación de los gestos para acciones del robot	53
Señales de referencia para la posición angular de cada grado de libertad	54
<i>Conversión de Matrices de rotación a ángulos de Tait-Bryan [58]</i>	55
<i>Ángulos de Tait-Bryan para el movimiento del Myo Armband</i>	56
Controladores PID en las articulaciones del brazo robótico de 3 GDL	59
<i>Calibración de los controladores PID</i>	60
<i>Calibración del primer grado de libertad (movimiento en yaw)</i>	62
<i>Calibración del segundo grado de libertad (movimiento en pitch)</i>	64
<i>Calibración del tercer grado de libertad (movimiento en roll)</i>	65
2.5 Implementación de la interfaz de usuario en Matlab	68
3. RESULTADOS Y DISCUSIÓN	70
3.1 Tasa de clasificación y Tiempo de respuesta del sistema de clasificación de gestos	70
3.2 Control de la posición angular del brazo robótico	74
4. CONCLUSIONES Y RECOMENDACIONES	77
4.1 Conclusiones	77
4.2 Recomendaciones	79
5. REFERENCIAS BIBLIOGRÁFICAS	81
6. ANEXOS	86

ANEXO I. SEÑALES ELECTROMIOGRÁFICAS.....	86
Señales EMG puras.....	86
Señales EMG posprocesadas.....	88
ANEXO II. INSTRUCCIONES DE ARMADO.....	91
Primer eslabón del brazo robótico.....	91
Segundo eslabón del brazo robótico.....	102
Base del brazo robótico.....	106
Pinza del brazo robótico.....	111
ANEXO III. MANUAL DE USUARIO.....	115
ANEXO IV. DIAGRAMAS DE FLUJO.....	117
ANEXO V. RUTINA DE PRUEBAS.....	121
Detalles de los usuarios.....	121
Matrices de confusión de cada usuario.....	121
Respuestas para diferentes entradas paso del primer grado de libertad (<i>yaw</i>)	126
Respuestas para diferentes entradas paso del segundo grado de libertad (<i>pitch</i>)	127
Respuestas para diferentes entradas paso del tercer grado de libertad (<i>roll</i>)..	128
ORDEN DE EMPASTADO.....	129

RESUMEN

En este trabajo se implementa un sistema de clasificación de gestos basado en las señales electromiográficas (EMG) del brazo adquiridas por el dispositivo comercial Myo Armband. A partir de este sistema de clasificación de gestos, se implementa el mando a distancia de un brazo robótico de 3 GDL.

Los gestos posibles de clasificar son los mismos gestos que reconoce el Myo Armband por defecto: mano en ángulo de 90° hacia adentro *Wave In*, mano en ángulo de 90° hacia afuera *Wave Out*, puño *Fist*, mano abierta *Open*, y pellizco con los dedos pulgar y medio *Pinch*.

Se utiliza una interfaz gráfica desarrollada en Matlab para adquirir las señales EMG correspondientes a cada gesto a clasificar. Cada gesto es repetido cinco veces en este entrenamiento. La clasificación se basa en el cálculo de la distancia DTW (*Dynamic Time Warping*) entre las señales EMG de un gesto desconocido con todas las señales EMG de entrenamiento. Para la clasificación de gestos se utiliza el algoritmo kNN (*k Nearest Neighbors*).

Para validar la tasa de clasificación del sistema y el tiempo de respuesta se realizaron pruebas con 10 voluntarios. A cada voluntario se le pidió realizar 30 repeticiones de cada gesto. En estas pruebas se consiguió una tasa de clasificación de 86% con un tiempo de respuesta menor a 250 milisegundos.

El brazo robótico de 3 GDL se implementó utilizando el kit Lego Mindstorms EV3. El gesto resultante de la clasificación, junto con las señales de la orientación del Myo Armband, son utilizados para el mando del brazo robótico. Se implementó un controlador PID sobre la posición angular de cada grado de libertad con un error en estado estable menor a 5°. La comunicación entre el brazo robótico y el computador se realizó vía Bluetooth.

PALABRAS CLAVE: EMG, clasificación, gestos, DTW, kNN, Mindstorms EV3, Myo Armband

ABSTRACT

This work is about the implementation of a hand gesture classification system based on the electromyographic signals obtained from the wearable Myo Armband. Based on this classification system, a 3 DOF robotic arm is operated remotely.

The gestures classified are Wave In, Wave Out, Fist, Open, Pinch, those are the default hand gestures that Myo Armband can recognize.

A user interface implemented in Matlab is used to acquire the EMG signals of all gestures to classify. The proposed model has a training routine where every gesture is repeated five times. Classification is the result of the DTW distance calculation between the EMG signals of an unknown gesture and all the gestures recorded during training. Finally, kNN classifier is used to determine the system's gesture prediction.

To validate the system classification accuracy and the system response time, tests with 10 volunteers were done. Each volunteer was asked to perform 30 repetitions for every gesture. From this test, the classification accuracy is 86% and classification accuracy and average response time of 210 milliseconds.

The 3 DOF robotic arm was implemented using the kit Lego Mindstorms EV3. The classified resulting gesture, altogether with Myo's orientation, are used to operate the robotic arm. A PID controller is applied over each degree of freedom assuring a steady state error lower than 5°. Communication with the robotic arm is via Bluetooth.

KEYWORDS: EMG, classification, gestures, kNN, DTW, Mindstorms EV3, Myo Armband

1. INTRODUCCIÓN

En los últimos años se ha desarrollado varios avances en la interacción hombre máquina. Esto debido a la creciente demanda por sistemas con mayores prestaciones, que sean capaces de aprovechar toda la versatilidad del movimiento del ser humano. Algunos de estos avances han dado como resultado varios dispositivos comerciales (Kinect, Leap Motion, Myo Armband) que son capaces de reconocer y clasificar un número limitado de movimientos y gestos. Si bien la mayoría de interfaces siguen utilizando consolas de control convencionales (como mouse y teclado), cada vez son más las aplicaciones que requieren de sistemas de clasificación y reconocimiento de gestos.

En la actualidad se desarrollan diversos sistemas de clasificación de gestos. Estos sistemas están siendo utilizados en el campo de la robótica, manufactura, medicina, aviación, etc. A pesar del amplio rango de aplicaciones, los sistemas de clasificación de gestos se diferencian principalmente, por el tipo de sensores que utilizan. La mayoría se hace uso de cámaras, sensores de flexión o sensores de actividad muscular [1] [2] [3]. De todos estos sensores, los sistemas de clasificación de gestos basados en las señales eléctricas de los músculos (EMG) son usualmente los de menor precio y mayor facilidad uso [4], lo que ha facilitado su investigación y desarrollo. Sin embargo, las investigaciones recientes demuestran que, ni la exactitud de clasificación, ni el tiempo de respuesta son suficientemente satisfactorios [1] [5] [6].

El presente Estudio Técnico implementa un sistema de clasificación de gestos competitivo, tanto en la exactitud de clasificación como en tiempo de respuesta, con los sistemas propuestos en la literatura científica. El sistema planteado realiza la clasificación de gestos basado en las señales EMG superficiales adquiridas por el dispositivo comercial Myo Armband. Como aplicación de este sistema de clasificación de gestos, un brazo robótico de 3 grados de libertad fue implementado con el kit Lego Mindstorms EV3. Para asegurar que la orden dada por el usuario sea ejecutada por el brazo robótico, se implementó un controlador PID de posición angular sobre cada grado de libertad del robot.

1.1 Objetivos

El objetivo general de este Estudio Técnico es:

IMPLEMENTAR UN SISTEMA DE CLASIFICACIÓN DE GESTOS DEL BRAZO HUMANO UTILIZANDO MYO ARMBAND PARA MANDO A DISTANCIA DE UN BRAZO ROBÓTICO DE 3 GDL.

Los objetivos específicos de este Estudio Técnico son:

- Obtener y almacenar las señales electromiográficas del antebrazo correspondientes a cada uno de los gestos a clasificar para su posterior análisis y clasificación.
- Implementar un algoritmo de clasificación utilizando las señales EMG medidas por el Myo Armband.
- Implementar un brazo robótico de 3GDL utilizando el kit Lego Mindstorms EV3.
- Implementar un mando a distancia para el brazo robótico de 3 GDL utilizando el sistema de clasificación de gestos y las señales de la IMU del Myo Armband.
- Realizar pruebas para determinar las tasas de clasificación y tiempos de procesamiento que permitan validar el funcionamiento del sistema integrado.

1.2 Alcances

El presente estudio técnico utilizará el sensor comercial Myo Armband para la medición de las señales EMG del antebrazo de un usuario. El actuador de este estudio técnico será un brazo robótico de 3GDL implementado con el kit Lego Mindstorms EV3.

Se implementará una interfaz gráfica en Matlab que permitirá a cada usuario entrenar el sistema de clasificación de gestos. Esta interfaz deberá mostrar una imagen al usuario de cada gesto a realizar durante el entrenamiento. Los gestos por clasificar serán: 1) mano en ángulo de 90° hacia adentro, 2) mano en ángulo de 90° hacia afuera, 3) puño, 4) pellizco con los dedos pulgar y dedo medio y 5) mano abierta.

Se implementará un algoritmo de clasificación de gestos. Por cada gesto a clasificar, el tiempo de respuesta del sistema será menor a 300ms. Adicionalmente, el sistema contará con una tasa de clasificación de al menos el 85%.

El brazo robótico será comandado a distancia utilizando los datos del Myo Armband. Los gestos clasificados serán usados como comandos de activación y desactivación del robot,

mientras que las señales de la IMU actuarán como referencia de los movimientos. Para el control de la posición angular de cada una de las articulaciones del robot, se implementará un controlador PID que asegure una precisión de 5°.

1.3 Marco Teórico

En esta sección se presenta una descripción del estado del arte de los sistemas de clasificación de gestos utilizando señales EMG, haciendo énfasis en sus falencias e introduciendo los métodos utilizados para este proyecto. Se divide el capítulo en cuatro apartados. En el primero se describe las señales electromiográficas (EMG), su naturaleza incluyendo un modelo de cómo se generan estas y de qué manera se manifiestan en los músculos del brazo. También se describen los métodos de adquisición de la señal EMG. En el segundo apartado se realiza una explicación del dispositivo comercial Myo Armband, dispositivo utilizado en este trabajo de titulación. Se detallan sus características técnicas, haciendo un análisis de sus ventajas y desventajas. En el tercer apartado se explican los conceptos básicos acerca de la clasificación de patrones, enumerando las principales técnicas utilizadas y su estructura. En el último apartado del marco teórico se describe brevemente algunas aplicaciones de los sistemas de clasificación gestos, analizando sus fortalezas y limitaciones, especialmente en el campo de la robótica. También presenta una comparación de los trabajos sobre clasificación de gestos para control robótico desarrollados en la Escuela Politécnica Nacional.

Descripción de las señales electromiográficas

Las señales electromiográficas son la base de varios sistemas de clasificación de gestos [1] [5] pues poseen información de los músculos encargados del movimiento. Estas señales eléctricas son el resultado de la actividad muscular y son producidas por comandos de movimiento que vienen desde el cerebro. Tener nociones básicas de su origen y naturaleza ayudará en el objetivo de diseñar un sistema de clasificación de gestos. A continuación, se presenta una breve descripción de la naturaleza de la señal electromiográfica.

Señal electromiográfica

Los músculos del brazo reciben los comandos para el movimiento desde el cerebro a través de impulsos eléctricos denominados acciones de potencial [7]. La medición de estas señales eléctricas se conoce como Electromiografía (de **electro**, electricidad; **mio**, músculo; y **grafía**, escritura), abreviado usualmente como EMG.

Las unidades motoras son responsables de la fuerza y el movimiento de los músculos. Un músculo está formado por varias unidades motoras, las cuales a su vez están formadas por fibras musculares. Cada unidad motora es inervada por una única motoneurona. Estas motoneuronas interconectan la médula espinal con las fibras musculares de una unidad motora. Cuando una motoneurona se activa, se propagan ondas, conocidas como trenes de acción de potencial, a lo largo de cada fibra [7] [8]. En la Figura 1.1 se muestra el esquema de una unidad motora que interconecta la médula espinal y las fibras musculares.

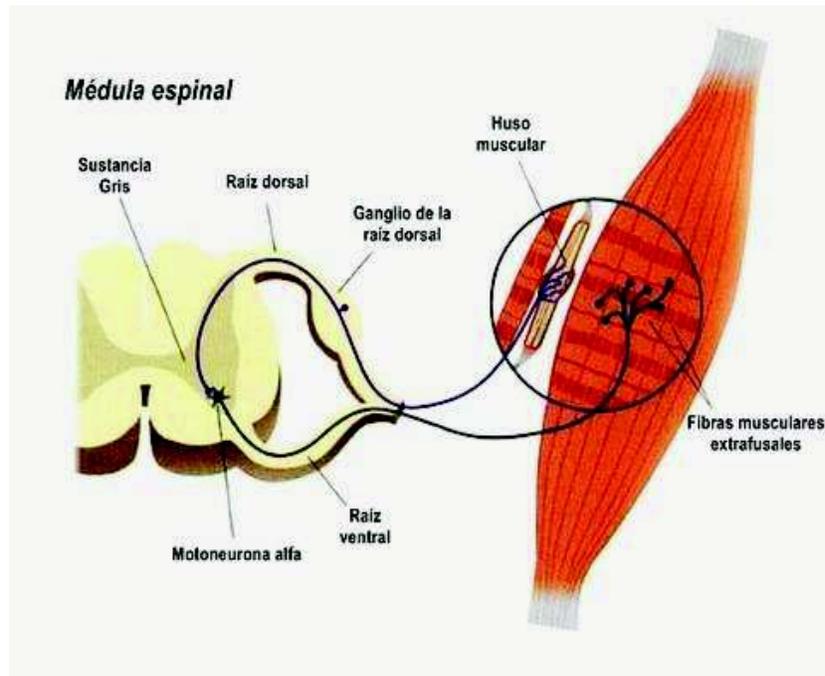


Figura 1.1 Unidad motora de un músculo [7]

Una acción de potencial de una unidad motora (MUAP por sus siglas en inglés, *motor unit action potential*) es un pulso eléctrico que viaja por una fibra muscular a intervalos irregulares con sus parámetros químicos característicos (propagación, conducción saltatoria, potencial de acción y reposo, etc). En la Figura 1.2 se muestra la forma de una acción de potencial de una unidad motora. El conjunto de señales eléctricas que viajan por una fibra muscular se conoce como tren de acciones de potencial de una unidad motora (MUAPT por sus siglas en inglés, *motor unit action potential train*).

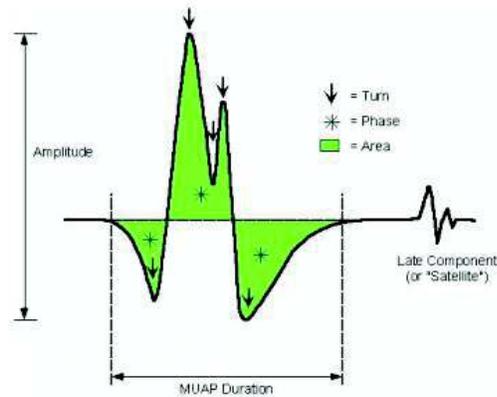


Figura 1.2 Acción de potencial de una unidad motora, MUAP [8]

Adquisición de la señal electromiográfica

Para la adquisición de las señales EMG se utilizan electrodos conectados a amplificadores diferenciales. Existen dos tipos de electrodos para medición de señales EMG: electrodos de aguja y electrodos de superficie. Los electrodos de aguja tienen una mejor precisión de medición, pues son insertados dentro del músculo a monitorear, a diferencia de los electrodos de superficie, que deben ser colocados en contacto con la piel [4]. Los electrodos de aguja tienen aplicaciones únicamente en el campo de la medicina debido a su gran precisión, pero no son utilizados en la industria, pues son considerados una medida invasiva. En cambio, los electrodos de superficie al ser una medida no invasiva son preferidos para aplicaciones generales [1] [4] [7].

Las señales EMG medidas con electrodos superficiales están en el orden de las decenas de milivoltios [9], por lo que se necesita una etapa de amplificación diferencial que eleve el voltaje a un nivel manejable, y que además elimine el ruido de la señal. En la Figura 1.3 se muestra la medición de la señal EMG usando 3 electrodos superficiales.

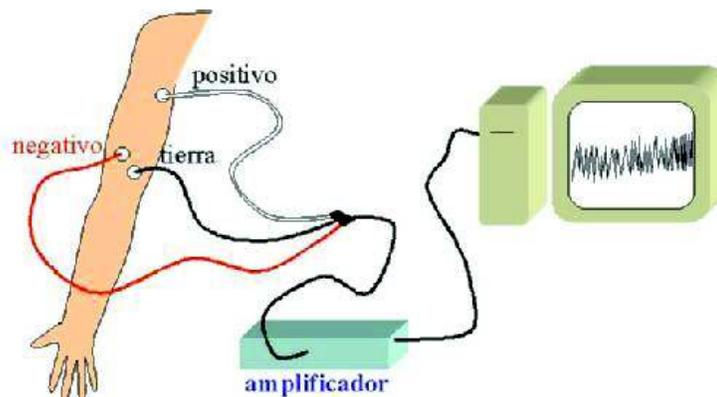


Figura 1.3 Medición de la señal EMG [9]

La señal EMG medida con electrodos superficiales, es la superposición de la señal eléctrica de todas las unidades motoras activas de uno o más músculos [10]. Puesto que cada músculo posee varias fibras musculares, y que los electrodos superficiales, inintencionalmente, captan la actividad de varios músculos a diferentes niveles de profundidad, la señal EMG resultante se compone de miles de pulsos eléctricos. En la Figura 1.4 se muestra un esquema de la descomposición de las señales EMG en sus trenes de acciones de potencial de una motora (MUAPT por sus siglas en inglés, *motor unit action potential trains*).

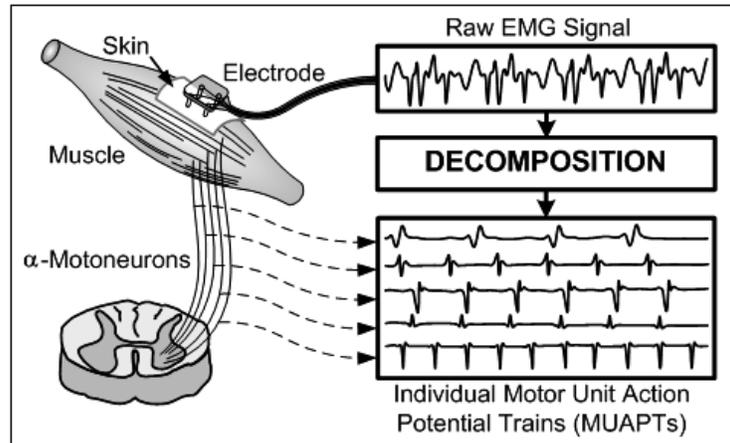


Figura 1.4 Descomposición de la señal EMG en unidades motoras [10]

La electromiografía es un campo donde todavía se realizan muchas investigaciones tanto para el campo médico como para el desarrollo de aplicaciones tecnológicas. En este proyecto, se utilizó el dispositivo Myo Armband, pues, al ser un dispositivo comercial, ya incluye el circuito de adquisición y amplificación de las señales EMG. Esto resulta en una ventaja para este proyecto, pues la manipulación y acondicionamiento de las señales de los electrodos es uno de los mayores inconvenientes que reportan varias investigaciones [11] [12].

Modelo matemático de la señal electromiográfica basado en la parte biológica

Un tren de acciones de potencial de una unidad motora, MUAPT, se representa como una serie de impulsos unitarios, $\delta(t)$, aplicados a la respuesta impulsiva $h(t)$. En este modelo, la señal EMG es la sumatoria de todas las MUAPTs de las unidades motoras dentro del músculo a medir [13], Figura 1.5. Este modelo se describe en las Ecuación 1.1, Ecuación 1.2 y Ecuación 1.3.

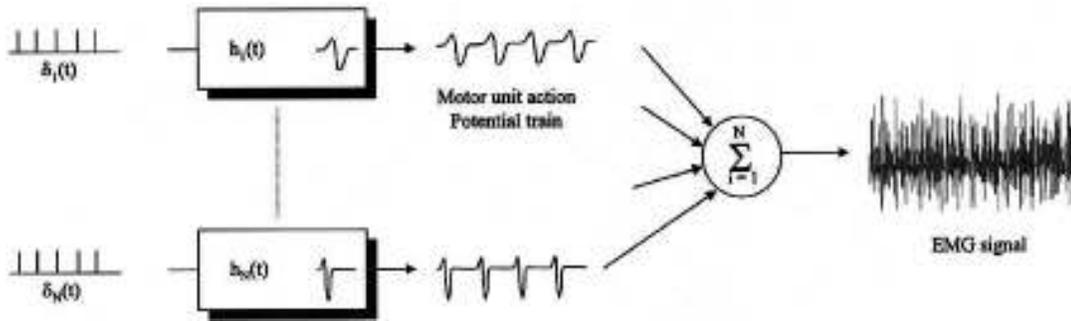


Figura 1.5 Modelo de la señal EMG basado en trenes de acción de potencial de una unidad motora [13]

$$\delta_n(t) = \sum_{k=1}^{m_n} \delta(t - t_{k_n})$$

Ecuación 1.1 Tren de impulsos unitarios

$$u_n(t) = \sum_{k=1}^{m_n} h_n(t - t_{k_n})$$

Ecuación 1.2 Tren de acciones de potencial de una unidad motora

$$EMG(t) = \sum_{n=1}^N u_n(t)$$

Ecuación 1.3 Señal EMG: suma de trenes de acciones de potencial

Donde: δ es la función delta de Dirac.

t_{k_n} , para $k = 1, 2, 3 \dots, m_n$, representa la ubicación temporal de los potenciales de acción de una unidad motora del n -ésimo tren.

m_n representa el número de acciones de potencial, MUAPs, en el n -ésimo tren, MUAPT.

h_n es la respuesta impulsiva de un sistema lineal invariante en el tiempo que representa a la n -ésima acción de potencial de la unidad motora (MUAP, ver Figura 1.2).

u_n es el n -ésimo tren de acciones de potencial de la unidad motora, MUAPT.

EMG es la señal electromiográfica medida en el instante t [13].

Modelo matemático de la señal electromiográfica basado en procesos gaussianos

Algunos autores [14] [15], en lugar de utilizar un modelo que descompone la señal EMG en sus trenes de acciones de potencial (sección 0), modelan a la señal electromiográfica como la suma de dos procesos gaussianos. El proceso $s(t)$ es modulado por la actividad muscular $m(t)$, e interferido por ruido aditivo blanco independiente, $n(t)$. En la Figura 1.6 se muestra el modelo de la señal EMG basado en este proceso. Los procesos, $s(t)$ y $n(t)$ se definen en la Ecuación 1.4. A partir de estos dos procesos se tiene el modelo de la señal electromiográfica de la Ecuación 1.5.

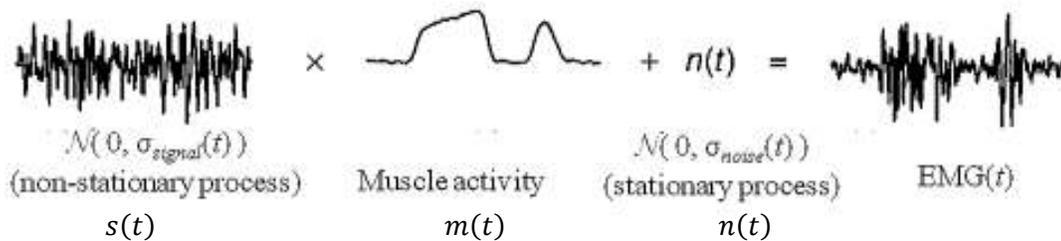


Figura 1.6 Modelo matemático de la señal EMG basado en procesos gaussianos

$$s(t) \in \mathcal{N}(0, \sigma_s(t)), \quad n(t) \in \mathcal{N}(0, \sigma_n)$$

Ecuación 1.4 Señal portadora y ruido blanco de la señal electromiográfica.

$$EMG(t) = s(t) * m(t) + n(t)$$

Ecuación 1.5 Modelo de la señal electromiográfica.

Donde: $\mathcal{N}(a, b)$ es un proceso gaussiano con media en a , desviación estándar b .

$EMG(t)$ es la medición de la señal electromiográfica en t

m es la actividad muscular

En este modelo se asume que el ruido, $n(t)$, es un proceso gaussiano estacionario, mientras que, $s(t)$, es no estacionario; ambos, además, independientes entre repeticiones [14] [15]. Un proceso estacionario representa a una variable aleatoria cuya función de distribución de probabilidad es constante para cualquier instante de tiempo [16]. Tal es el caso de $n(t)$, donde σ_n es constante. En cambio, un proceso no estacionario es aquel donde la media y/o la varianza dependen del tiempo, debido generalmente a una acumulación de materia o energía [16]. La señal portadora $s(t)$, es un proceso no estacionario, pues su desviación estándar $\sigma_s(t)$ varía en el tiempo.

Myo Armband

El Myo Armband es un dispositivo para reconocimiento de gestos distribuido en el mercado desde 2014 por la empresa canadiense Thalmic Labs Inc [17]. El Myo Armband tiene forma de brazalete con ocho electrodos diferenciales de superficie; además, posee una unidad de medición inercial (IMU) de 9 grados de libertad (acelerómetro, magnetómetro y giroscopio en x, y, z). El Myo Armband se muestra en la Figura 1.7.



Figura 1.7 Myo Armband™ desarrollado por Thalmic Labs Inc. [17]

Características del Myo Armband

Las principales características del Myo Armband son:

- 8 electrodos diferenciales de superficie.
- IMU (acelerómetro, magnetómetro y giroscopio en x, y, z).
- Frecuencia de medición del EMG: 200 Hz.
- Frecuencia de medición de la IMU: 50 Hz.
- Sistema propietario para el reconocimiento de cinco gestos: mano en ángulo de 90° hacia adentro *Wave In*, mano en ángulo de 90° hacia afuera *Wave Out*, puño *Fist*, mano abierta *Open*, y pellizco con los dedos pulgar y medio *Pinch*.
- Comunicación Bluetooth®.
- Batería recargable de ion litio de larga duración.
- Realimentación háptica a través de vibraciones [18].

El Myo Armband posee la integración entre los sensores y el circuito de amplificación, por lo que no es necesario preocuparse de la adquisición de las señales. Además, el Myo Armband es un dispositivo inalámbrico que permite una libre movilidad del brazo en que se lo utilice; esto lo vuelve atractivo para ser la base de muchas aplicaciones. El Myo Armband también posee integrada una batería recargable. A diferencia de otros dispositivos comerciales (drones, celulares, etc), esta batería tiene una larga duración, lo que permite largas sesiones de pruebas sin interrupción.

Sistemas de clasificación de gestos

La clasificación de gestos es un problema del área de Reconocimiento de Patrones y Aprendizaje Computacional, que consiste en la interpretación matemática de gestos significativos a través de una computadora [19]. Un gesto es una forma de comunicación no verbal compuesta de movimientos visibles del cuerpo para comunicar mensajes particulares. Estos mensajes pueden ser faciales, con las manos, o cualquier otra parte del cuerpo [20]. En la Figura 1.8 se muestra el gesto mano abierta u *Open*.



Figura 1.8 Gesto mano abierta [18]

Existen varias investigaciones recientes enfocadas en la clasificación de gestos. Todas estas investigaciones se enfocan en mejorar las tasas de clasificación y reducir el tiempo de procesamiento [1] [21]. A continuación, se realiza la descripción del problema de clasificación y sus etapas.

Etapas de los sistemas de clasificación de gestos

Un problema de clasificación en Reconocimiento de Patrones puede ser dividido en cinco etapas. Estas etapas son las siguientes:

- *Adquisición*: medios en que la señal es adquirida utilizando un sensor específico.
- *Preprocesamiento*: la señal adquirida del sensor es procesada para eliminar cualquier presencia de ruido.
- *Extracción de características*: características específicas de la señal son extraídas para su posterior análisis.

- *Clasificación*: las señales a clasificar son identificadas con la categoría correspondiente.
- *Postprocesamiento*: los resultados de la clasificación son procesados para poder ser utilizados en alguna aplicación.

Adquisición de señales

La adquisición de señales es el proceso de medir un fenómeno físico a través de sensores para su posterior utilización en una computadora [22]. Los sensores utilizados para la adquisición de señales dependen de la variable física a medir. En la figura se muestra un esquema típico de un proceso de adquisición de señales.



Figura 1.9 Esquema típico de un proceso de adquisición de señales [22]

Los sistemas de clasificación de gestos se basan principalmente en los datos extraídos mediante cámaras, sensores de flexión o señales EMG [2] [3] [23]. Al utilizarse cámaras para la clasificación, se consigue la ventaja de no intervenir con el usuario, pero presentan problemas ante variaciones de luz, oclusión y variación de distancia con respecto a la cámara [2]. En el caso de los sensores de flexión, estos se instalan en un guante para que lo utilice el usuario, lo que puede ocasionar molestias, a pesar de conseguir altas tasas de clasificación [3]. Finalmente, los sistemas de clasificación de gestos que utilizan señales EMG presentan la ventaja de ser de bajo costo y de fácil uso. En la Figura 1.10 se muestra la colocación del brazalete Myo Armband para clasificación de gestos.



Figura 1.10 Uso del brazalete Myo Armband [18]

Es importante notar que los sistemas de clasificación de gestos no se limitan en su reconocimiento a una sola variable física, sino que pueden hacer clasificaciones basadas en múltiples señales de distinta naturaleza. Este es el caso del Myo Armband, que utiliza conjuntamente las señales electromiográficas con las señales del magnetómetro, giroscopio y acelerómetro. En el presente trabajo, se utiliza las señales electromiográficas para implementar un sistema de clasificación de gestos, y se utiliza las señales de aceleración para el mando del brazo robótico.

Extracción de características

Los sistemas de clasificación de gestos predicen a qué categoría pertenece una señal desconocida al compararla con un conjunto de señales de entrenamiento de las que se conoce a qué categorías pertenecen [24] [25]. Esta predicción (clasificación) es realizada a partir de un vector de características que representa la señal. Un vector de característica es un vector que contiene información distintiva y no redundante de la señal en una dimensión menor a la señal original [26]. Resulta conveniente utilizar un vector de características para la clasificación de una señal, pues al ser de menor dimensión que la señal, ahorra tiempo de procesamiento. Esta extracción de características se realiza con el objetivo de eliminar información de la señal no relevante para la clasificación. En la Figura 1.11 se muestra el esquema de un sistema de clasificación con la etapa de adquisición de señales, extracción de características y clasificación. También se añade un bloque que ilustra las aplicaciones que este tipo de sistemas pueden tener.

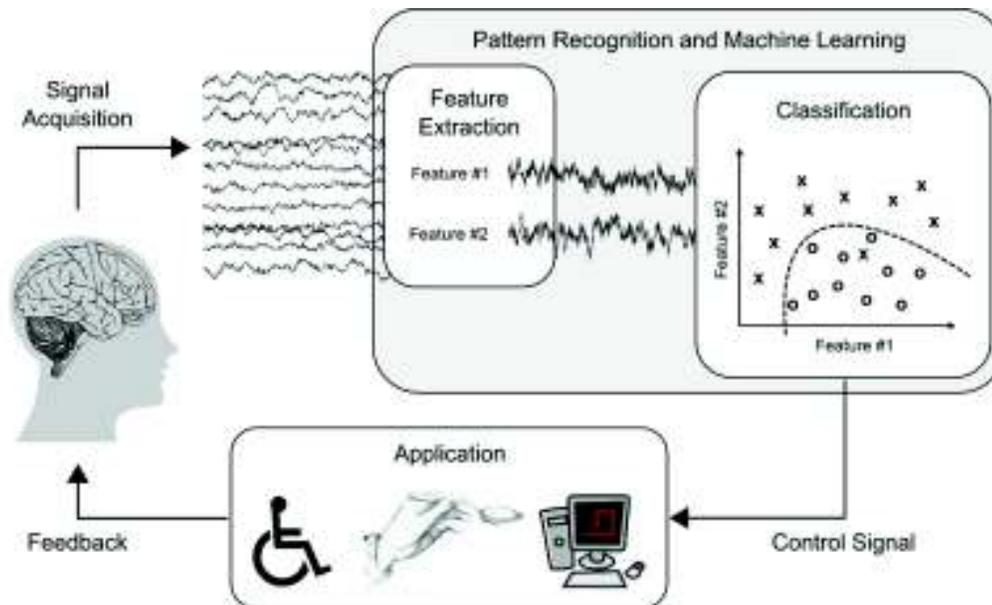


Figura 1.11 Sistema de clasificación con aplicación: adquisición de señales, extracción de características, clasificación, señal de control y aplicación [26]

Varias investigaciones enfocadas en la clasificación de gestos utilizan generalmente las siguientes características: valor medio, desviación estándar, covarianza, valor eficaz (entre otras) como características [21] [27] [28] [29]. El escoger entre uno u otro conjunto de características puede variar la exactitud de un clasificador. Por lo tanto, en la mayoría de los sistemas de clasificación, el escoger un buen conjunto de características es indispensable para obtener buenos resultados.

Clasificadores

Los clasificadores son algoritmos utilizados en Aprendizaje Computacional (ML por sus siglas en inglés, Machine Learning) que permiten identificar, separar, o asignar una señal a una clase perteneciente dentro de un conjunto determinado de clases [24] [30].

Los clasificadores que conocen de antemano la etiqueta correspondiente a cada señal son conocidos clasificadores de aprendizaje supervisado. En cambio, cuando el sistema no posee información de la clase a la que corresponden las señales, es necesario que el algoritmo infiera las clases por su cuenta; este estilo de aprendizaje se conoce como aprendizaje no supervisado [31]. En la Figura 1.12 se ve un esquema del aprendizaje supervisado y no supervisado. El presente trabajo desarrolla un sistema de clasificación de gestos con aprendizaje supervisado.

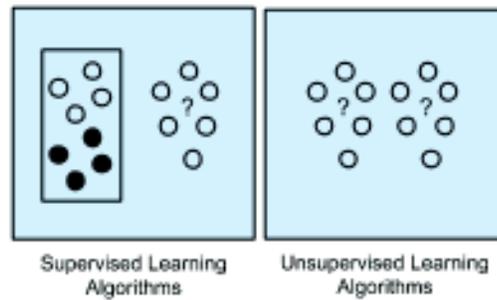


Figura 1.12 Aprendizaje Supervisado y No Supervisado [31]

La clasificación de gestos es un procedimiento, generalmente, de aprendizaje supervisado, pues se conoce previamente la categoría a la que corresponde cada señal [25]. Entre los clasificadores más comunes para la clasificación de gestos tenemos k vecinos más cercanos (kNN, por sus siglas en inglés, k Nearest Neighbors), Máquinas de Vectores de Soporte (SVM, Support Vector Machines), Modelos Ocultos de Márkov (HMM, Hidden Markov Models) y Redes Neuronales Artificiales (ANN, Artificial Neural Networks) y árboles de decisión [24] [26]. En la Figura 1.13 se presenta las fronteras de decisión implementadas usando los clasificadores Árboles de Decisión, Maquinas de Vectores de Soporte, Redes Neuronales y Bosques Aleatorios para un sistema de tres clases.

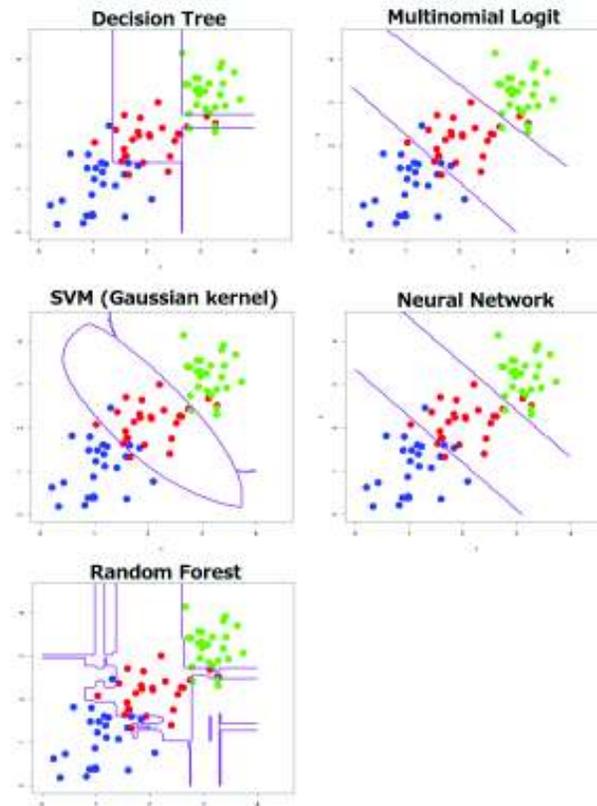


Figura 1.13 Fronteras de decisión de un problema con 3 clases [26]

Aplicaciones de los sistemas de clasificación de gestos

Como parte de las investigaciones acerca de clasificación de gestos, también existen varios trabajos enfocados en desarrollar aplicaciones basadas en estos sistemas. Estas aplicaciones se realizan en el campo de la robótica, del entretenimiento (desarrollo de interfaces para videojuegos), interacción hombre máquina, entre otras [21] [27] [28].

Para el caso del dispositivo comercial Myo Armband, se encuentran disponibles varias aplicaciones en línea. Estas aplicaciones cubren un amplio rango; por ejemplo, existen aplicaciones para presentación de diapositivas (PowerPoint), control de drones (Parrot 3.0), mando de videojuegos (Fruit Ninja), programación (Arduino), control multimedia (Spotify, YouTube), etc [32]. Estas aplicaciones comerciales junto con las desarrolladas en las investigaciones, aunque novedosas, todavía tienen grandes limitaciones pues carecen de la precisión y naturalidad necesarias [1].

En el presente estudio técnico se incluye una aplicación robótica para el sistema de clasificación de gestos utilizando Myo Armband.

Sistemas de clasificación de gestos aplicados a la robótica

Uno de los mayores campos de aplicación de la clasificación de gestos es la robótica. Xu et al. [33] desarrollaron un sistema utilizando dos brazaletes Myo, uno en el antebrazo y otro a la altura del brazo. Aunque pueden detectar con gran precisión el movimiento del brazo, su sistema no realiza el reconocimiento de ningún gesto de la mano. En la Politécnica Nacional existen dos trabajos que incluyen clasificación de gestos en base a señales electromiográficas. En la investigación de Hidalgo et al [12], se plantea el uso de las señales EMG para el control difuso de un brazo robótico. Además del trabajo de Hidalgo et al., existe también el trabajo de Gallegos y Simbaña que incluye el control de una pinza robótica de dos grados de libertad utilizando cinco electrodos de superficie [11]. Este trabajo realiza el control de una pinza robótica de limitadas prestaciones, principalmente porque cuenta con pocos gestos reconocidos, solo dos. En estos dos trabajos vemos que el número de gestos reconocidos es muy bajo, además de que el brazo robótico que implementan posee una movilidad limitada. Todos estos ejemplos evidencian que todavía no se ha implementado exitosamente una aplicación para un sistema de clasificación de gestos que pueda utilizarse de manera natural e intuitiva.

En el presente trabajo se desarrolla una aplicación robótica, la misma que consiste en un brazo de 3 grados de libertad implementado utilizando Lego Mindstorms.

2. METODOLOGÍA

El presente estudio técnico es una investigación del tipo experimental pues a través de experimentos se determina el cumplimiento de los objetivos planteados. Los datos utilizados se obtienen a través de mediciones; para el sistema de clasificación de gestos se recolecta la señal electromiográfica de un grupo de estudio; mientras que, para el brazo robótico, se realizan mediciones de posición angular sobre cada grado de libertad.

En este capítulo se describe las características del sistema de clasificación de gestos implementado, así como también se describe el mando a distancia del brazo robótico de 3 grados de libertad y la interfaz gráfica implementada.

Las secciones correspondientes al sistema de clasificación de gestos tratan de la adquisición de datos, la fase de entrenamiento y la fase de clasificación. En la sección 2.1 se describe la adquisición de datos de la señal EMG, la librería utilizada y la ubicación recomendada del Myo Armband en el brazo. La sección 0 se trata sobre la fase de entrenamiento; aquí se define los gestos a clasificar en este proyecto.

En la sección 2.3 se describe el sistema de clasificación: los detalles del preprocesamiento de la señal EMG, el algoritmo de alineamiento temporal dinámico (DTW por sus siglas en inglés, *Dynamic Time Warping*) utilizado como métrica de distancia, y el clasificador utilizado, kNN.

La sección 2.4 trata sobre el brazo robótico de tres grados de libertad implementado. Su implementación con el kit Lego Mindstorms, los controles PID en cada grado de libertad y el método utilizado para su calibración. La sección 2.5, última de este capítulo, es sobre la interfaz gráfica implementada en Matlab.

2.1 Adquisición de datos del Myo Armband

Para la comunicación del computador con el Myo Armband se requiere la instalación del software Myo Connect [34]. También es necesaria la instalación del Myo SDK 0.9.0. Esta es la librería oficial para el desarrollo de aplicaciones con el Myo Armband [35]. En el presente estudio técnico se utiliza, además, la librería “MyoMex” de código abierto disponible en Github desarrollada por Mark Tomaszewski [36]. Esta librería permite adquirir los datos del Myo Armband para ser utilizados en Matlab. La estructura de adquisición de datos se muestra en la Figura 2.1.

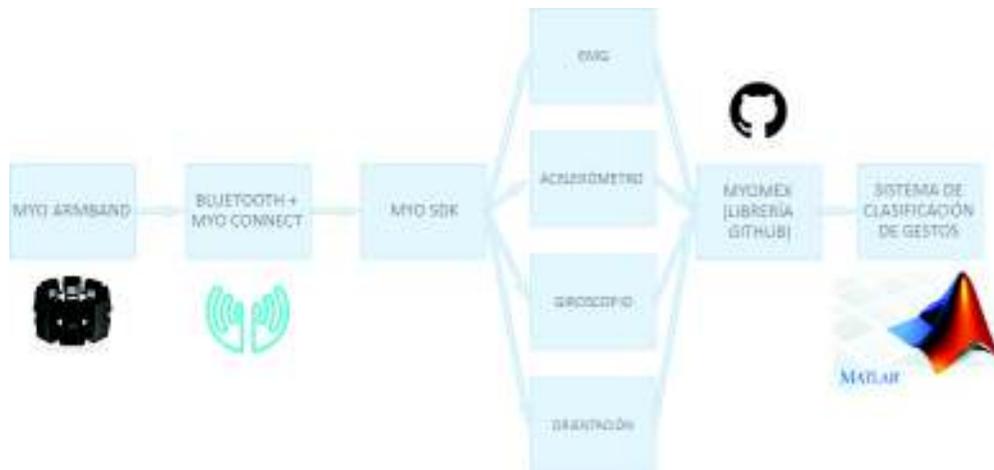


Figura 2.1 Adquisición de datos del Myo Armband hacia Matlab

Librería MyoMex

La librería MyoMex incluye ciertas características propias con respecto al Myo Armband. Esta librería realiza el muestreo de datos cada 100 milisegundos. Todas las señales de tiempo se almacenan en un buffer. La librería MyoMex, además, realiza una normalización de las señales electromiográficas para que estén en el rango de -1 a 1 (el factor de escala es 128). MyoMex también incluye a la matriz de rotación, esta matriz de rotación es la utilizada como señal de orientación del brazo en este proyecto. En la Tabla 2.1 se enumeran las variables disponibles de la librería MyoMex para uso del usuario con sus unidades respectivas.

Tabla 2.1 Datos disponibles de la librería MyoMex

Variable		Unidad
<i>cuaternio de orientación</i>	$[w, x, y, z]$	[unitario]
<i>giroscopio</i>	$[x, y, z]$	$\left[\frac{rad}{seg^2} \right]$
<i>acelerómetro</i>	$[x, y, z]$	$[g] = \left[9.8 \frac{m}{s^2} \right]$
<i>EMG</i>	$[1 \times 8]$	[unitario]
<i>Matriz de rotación</i>	$[3 \times 3]$	[unitario]
<i>Frecuencia del EMG</i>	200	[Hz]
<i>Frecuencia de la IMU</i>	50	[Hz]

Colocación del Myo Armband

El Myo Armband es un brazalete portable que el usuario debe colocarse en contacto con la piel. Debido a esta portabilidad, es posible, que los electrodos se ubiquen en posiciones diferentes en cada ocasión. En general, se recomienda a los usuarios utilizar el Myo Armband siempre en la misma posición del antebrazo. Esta posición es con el logotipo del Myo Armband apuntando hacia arriba, con el brazo paralelo al suelo y la palma hacia abajo. Colocando el brazalete en la parte más ancha del antebrazo se evita que este se deslice. Esta posición busca maximizar la conexión eléctrica entre los electrodos y la piel para un mejor rendimiento [37]. En la Figura 2.2 se muestra una fotografía de la ubicación idónea del Myo Armband.



Figura 2.2 Ubicación del Myo Armband en el antebrazo

Representación de la Señal EMG medida por el Myo Armband

Las señales EMG adquiridas utilizando el Myo Armband y la librería MyoMex permiten realizar un análisis en Matlab. En la Figura 2.3 se muestra un ejemplo de la señal electromiográfica medida por el Myo Armband.

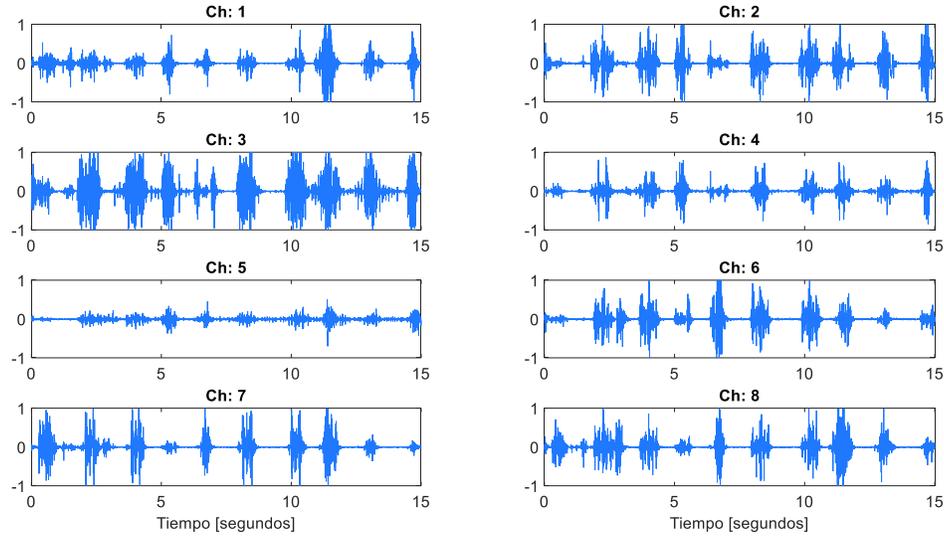


Figura 2.3 Señales EMG adquiridas por los 8 canales del Myo Armband

Se observa que, en los instantes de tiempo donde existe actividad muscular, la señal EMG tiene una amplitud mayor. Esta actividad muscular tiene diferente amplitud para cada canal dependiendo del tipo de movimiento. Además, cuando el brazo está en reposo, la señal se encuentra oscilando alrededor del cero.

La señal EMG adquirida está formada por mediciones discretas cada T segundos. De la sección 0 se sabe que el periodo de muestreo, T , de la señal EMG es 200 ms. Por lo tanto, la señal EMG medida en una ventana de tiempo, $V|_{n-c-1}^n$, contiene c muestras discretas. Esta señal EMG, W , se representa matemáticamente en la Ecuación 2.1.

$$W = [E_{(n-c)-1}, \dots, E_{(n-1)}, E_{(n)}]^T$$

Ecuación 2.1 Señal EMG dentro de la ventana de tiempo con c muestras

Donde: $n = \frac{t}{T}$ es un entero que representa un instante discreto del tiempo t .

$E_{(n)} = [EMG_{1(n)} \dots EMG_{8(n)}]$ con $EMG_{i(n)}$ siendo la n -ésima muestra de la señal EMG del canal i

2.2 Entrenamiento del sistema de clasificación de gestos

Como se vio en el apartado 0 sobre los sistemas de clasificación, un algoritmo de Aprendizaje Supervisado requiere datos de aprendizaje que incluyan información sobre la categoría a la que pertenecen. En la mayoría de los casos, es necesario recopilar esta información en un procedimiento previo a la clasificación [31]. A esta etapa se la conoce como etapa de entrenamiento. Los parámetros de esta etapa son el número de gestos a reconocer y las repeticiones de cada gesto dentro de una ventana de tiempo.

En el presente estudio técnico se propone que la rutina de entrenamiento se realice por cada usuario que desea utilizar el sistema; de esta manera se reduce la complejidad del problema de clasificación. El entrenamiento consiste en adquirir las señales electromiográficas del antebrazo a través del Myo Armband. Se incluye 5 clases de gestos en la rutina de entrenamiento, las mismas 5 clases de gestos que reconoce el Myo Armband por defecto. Cada usuario debe realizar 5 repeticiones por cada gesto a reconocer. El usuario tiene una ventana de 2 segundos para realizar cada repetición. En la Tabla 2.2 se resumen estos parámetros de la etapa de entrenamiento.

Tabla 2.2 Parámetros de la rutina de entrenamiento

Parámetro de entrenamiento		
<i>Número de clases de gestos</i>	5	[gestos]
<i>Número de repeticiones por clase de gesto</i>	5	[repeticiones]
<i>Tiempo de cada repetición</i>	2	[segundos]

Se escogió una ventana de 2 segundos para cada repetición pues de las señales EMG adquiridas se observó que ningún gesto tiene una duración mayor a este tiempo. De hecho, 2 segundos es una ventana de tiempo holgada para que el usuario realice el gesto. De esta manera, no se pierde fragmentos de la señal por distracción del usuario. El número de repeticiones del entrenamiento se escogió heurísticamente, teniendo en cuenta los parámetros del clasificador a utilizarse (que se describe más adelante).

Definición de los gestos para la clasificación

El Sistema de clasificación de gestos propuesto en este trabajo permite identificar 5 gestos de la mano. Los 5 gestos son definidos partiendo de la posición de reposo, y terminan cuando se regresa a la misma posición. Los gestos, x_i con $i = 1, 2, \dots, 5$, son los mismos gestos que puede reconocer por defecto el sistema propietario del Myo Armband. Estos

gestos fueron elegidos para el proyecto Junior “Clasificación de señales electromiográficas del brazo humano usando técnicas de reconocimiento de patrones y Machine Learning” que busca implementar un sistema de clasificación de gestos en tiempo real y con alta exactitud de clasificación. En la Figura 2.4 se muestra una imagen de cada uno de los gestos.



Figura 2.4 Gestos del sistema de clasificación

Puño (x_1 Fist): desde la posición de reposo, los dedos de la mano deben cerrarse con un poco de fuerza, después de aproximadamente un segundo se regresa a la posición inicial.

Mano 90° hacia adentro (x_2 Wave In): con los dedos apretados y alargados, mover toda la mano desde la muñeca hacia adentro 90°. Termina el gesto, al regresar a la posición de reposo.

Mano 90° hacia adentro (x_3 Wave Out): con los dedos apretados y alargados, mover 90°, en este caso hacia afuera. Se debe regresar a la posición inicial del mismo modo.

Mano abierta (x_4 Open): se aplica fuerza en los dedos tratando de extenderlos lo más posible, manteniendo a la mano en la misma dirección del antebrazo. Para terminar, se relajan los dedos hasta la posición inicial.

Pellizco (x_5 Pinch): se realizan dos golpes rápidos entre los dedos pulgar y medio.

Para la rutina de entrenamiento se incluyó además un sexto gesto: ***posición de reposo*** (o también conocido como “no Gesto”) La posición de reposo, x_0 , es con el antebrazo relajado, horizontal, 90° con respecto al brazo. Este gesto es incluido para determinar los momentos en el que la mano no se encuentre en movimiento. El gesto de reposo también sirve como fondo para el clasificador, pues el sistema no posee datos del conjunto total de gestos posibles que un usuario puede realizar. A partir de este punto, se hará mención a cada gesto por su nombre en inglés, pues estos son nombres propios.

Señales Electromiográficas de los gestos de entrenamiento

En la Figura 2.5 se muestra una comparación las señales EMG de las cinco clases de gestos por canal. Como se definió en el apartado anterior (sección 0), un gesto empieza y termina en la posición de reposo.

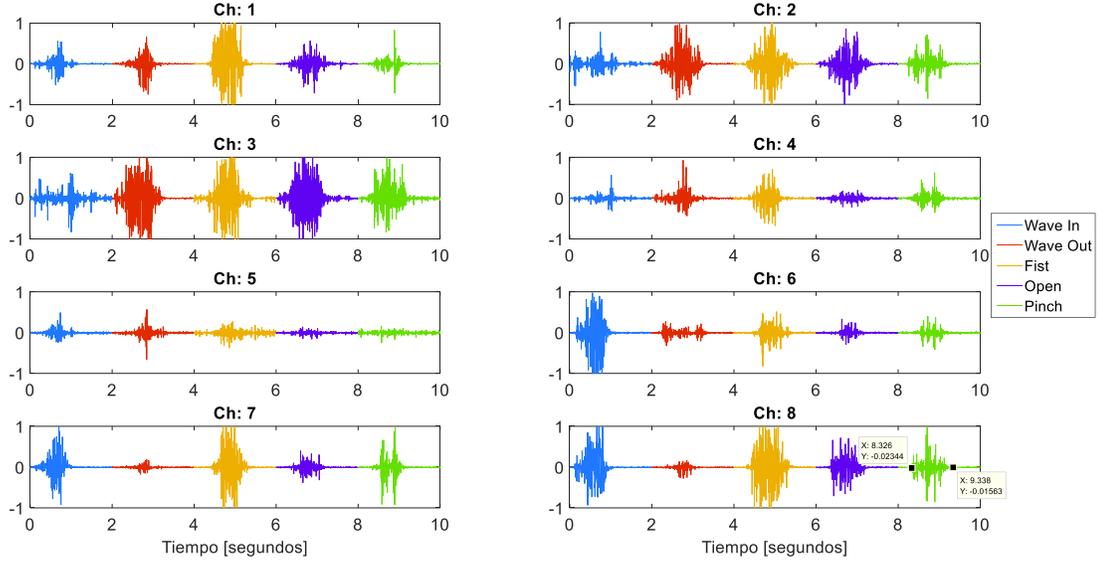


Figura 2.5 Señales EMG de los 5 gestos del sistema de clasificación de gestos

Vemos que entre el gesto *Wave In* y *Wave Out*, la actividad muscular se da en canales opuestos; mientras *Wave In* activa los canales del 6 al 8, el gesto *Wave Out* activa los canales 2 y 3. Esto debido al hecho de que son gestos que mueven en direcciones opuestas a la mano. De manera similar, se observa que las señales EMG del gesto *Fist* tienen mayor amplitud que las del gesto *Open* en los canales 1, 4 y 7. También se observa que la duración de todos los gestos no excede los 2 segundos. El gesto más corto en duración es el gesto pellizco o *Pinch* (alrededor de 0.8 segundos). Por lo tanto, al seleccionar una ventana de dos segundos, se asegura que el usuario tiene un margen suficiente de tiempo para que la señal EMG no sea recortada.

En el ANEXO I. SEÑALES ELECTROMIOGRÁFICAS se presenta las figuras correspondientes a las señales EMG para cada uno de los gestos.

Todas las señales EMG, (representadas con W , Ecuación 2.1) grabadas durante 2 segundos, equivalente a 400 muestras $V|_0^{399}$, forman la base de datos $B \in \mathbb{R}^{[400 \times 8 \times 30]}$. En la Ecuación 2.2 se define la base de datos B .

$$B = [W_1, W_2, \dots, W_{30}]^T$$

Ecuación 2.2 Base de datos B formada por las señales EMG W

Donde: W_m es la señal EMG dentro de una ventana tiempo $V|_0^{399}$ correspondiente al m gesto.

m es la repetición del entrenamiento, $m \in [1,2, \dots, 30]$, tal que $[1,2 \dots, 5]$ corresponde al gesto x_1 del entrenamiento, $[6,7 \dots, 10]$ corresponde al gesto x_2 , y así sucesivamente.

2.3 Clasificación de gestos en tiempo real

La clasificación de gestos en tiempo real requiere que el sistema retorne resultados por lo menos cada 300 ms [6]. El sistema de clasificación de gestos, por lo tanto, debe devolver una predicción, y , para cada señal EMG desconocida, W , en cada iteración N (donde el tiempo entre la iteración N y $N - 1$ debe ser menor a 300 ms). Puesto que los gestos definidos en la sección 0. tienen una duración alrededor de un segundo, se define una ventana de tiempo que limita el tamaño de W a las últimas 200 muestras de la señal EMG.

La señal EMG correspondiente a un gesto desconocido dentro de una ventana de tiempo de 1 segundo (200 muestras) se representa como $W(N)$:

$$W(N) = [E_{(n-199)}, E_{(n-198)}, \dots, E_{(n-1)}, E_{(n)}]^T$$

Ecuación 2.3 Señal EMG desconocida dentro de la ventana de tiempo de 200 muestras

Donde: $E_{(n)} = [EMG_{1(n)} \dots EMG_{8(n)}]$ fue definido en la Ecuación 2.1

N es un entero positivo que representa la n ésima iteración del sistema de clasificación.

La clasificación en tiempo real genera ventanas consecutivas $W(1), W(2), \dots, W(N)$. El tiempo, ξ , transcurrido entre dos ventanas consecutivas $W(N)$ y $W(N - 1)$ presenta un solapamiento. En la Figura 2.6 se observa múltiples ventanas de tiempo, W , de la señal EMG donde el solapamiento es del 75%.

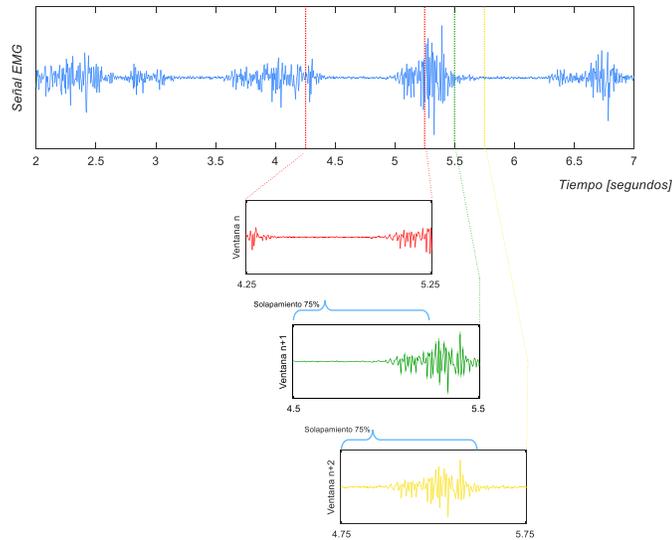


Figura 2.6 Ventana de tiempo $W(N)$

La clasificación en tiempo real añade dos parámetros al sistema de clasificación de gestos: la duración de la ventana de tiempo, y el tiempo entre dos ventanas consecutivas ξ . Para el presente proyecto, se escogió una duración de la ventana de tiempo, equivalente a un segundo de señal EMG (200 muestras), pues, de la etapa de adquisición de datos y entrenamiento, se observa que ningún gesto excede este tiempo. El tiempo de salto de la ventana, ξ , está limitado al tiempo de respuesta del sistema. Al realizar varias pruebas se llegó a establecer el salto de la ventana, ξ , en 200 ms. Estos parámetros se resumen en la Tabla 2.3.

Tabla 2.3 Parámetros para la clasificación en tiempo real

Parámetros para la clasificación en tiempo real		
<i>Duración de la ventana de tiempo</i>	1	[s]
<i>Tiempo de salto de la ventana, ξ</i>	200	[ms]

Preprocesamiento de la señal electromiográfica

Las señales electromiográficas, antes de la clasificación, deben ser pre-procesadas en concordancia con el clasificador elegido. Este preprocesamiento de la señal se realiza tanto para una señal desconocida, $W(N)$, como para las señales de la base de datos adquiridas en el entrenamiento B (Ecuación 2.2).

Como se mencionó en el apartado 0, las técnicas utilizadas para el procesamiento de la señal EMG son muy variadas, y dependen del clasificador a utilizarse. En el presente estudio técnico se utiliza el clasificador kNN con la distancia del algoritmo DTW (alineamiento

temporal dinámico). El preprocesamiento por aplicar sobre la señal electromiográfica, por ende, se establece considerando las características del algoritmo DTW (que se explicará en secciones más adelante).

El modelo matemático de la señal electromiográfica (sección 0) establece que la señal EMG está compuesta por una señal portadora de media cero modulada por una señal que contiene la actividad muscular. El objetivo de la etapa de preprocesamiento es eliminar la componente gaussiana de ruido blanco, y trabajar únicamente con las señales modulantes de muy baja frecuencia que contienen información de la actividad muscular. Para el presente estudio técnico, las señales EMG de los ocho canales del Myo Armband, son rectificadas y filtradas aplicando un filtro pasa bajos de Butterworth.

Descripción del Filtro de Butterworth

El filtro de Butterworth es un filtro que se caracteriza por tener una respuesta en magnitud plana hasta la frecuencia de corte [38]. En la Ecuación 2.4 se presenta la respuesta de un filtro de Butterworth de enésimo orden.

$$|H(j\omega)|^2 = \frac{1}{1 + \left(\frac{j\omega}{j\omega_c}\right)^{2N}}$$

Ecuación 2.4 Ganancia del filtro de Butterworth de orden N .

Donde: ω_c es la frecuencia de corte y N el orden del filtro.

El filtro de Butterworth, a partir de la frecuencia de corte, tiene una respuesta que disminuye a razón de $20N \text{ dB}$ por década [39]. A la frecuencia de corte, el filtro de Butterworth tiene una ganancia de -3dB . En la Figura 2.7 se presenta la respuesta en magnitud del filtro de Butterworth.

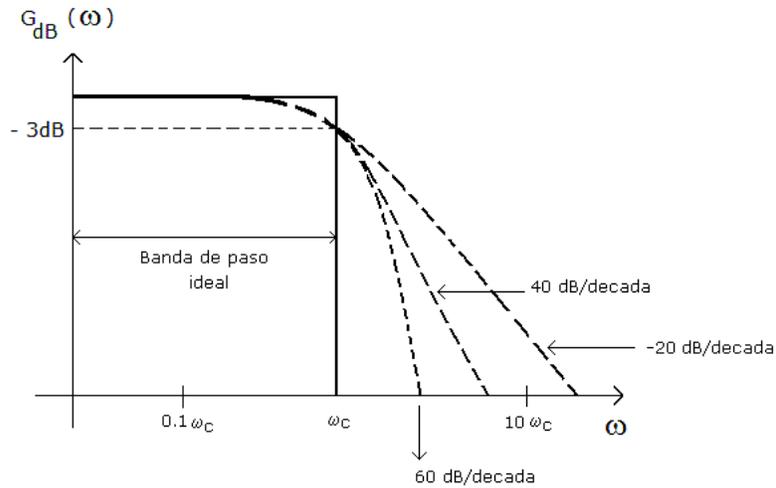


Figura 2.7 Filtro de Butterworth de primer (-20 dB/decada), segundo (-40 dB/decada) y tercer orden (-60 dB/decada) [39]

Rectificación y filtrado de las señales EMG

Es necesario aplicar el mismo preprocesamiento a las señales EMG de la base de datos, B , que a la señal EMG desconocida $W(N)$. La señal procesada, ψ , de la señal desconocida EMG, $W(N)$, resulta al aplicar la operación de filtrado, \mathcal{F} , y rectificado, abs .

$$\psi_i = \mathcal{F}(abs(e_i), \omega_c)$$

Ecuación 2.5 Preprocesamiento del i ésimo canal de la señal EMG

Donde: $\mathcal{F}(x, \omega)$ es el operador que aplica el filtro de Butterworth sobre x , con frecuencia de corte normalizada ω_c .

ψ_i es la señal EMG procesada correspondiente al canal i

$e_i \in W(N)$ es el conjunto de todas las muestras del canal i de la señal EMG desconocida, $W(N)$ (Ecuación 2.1).

En la Ecuación 2.6 se define al conjunto de señales procesadas ψ_i para la n ésima iteración N .

$$\Psi(N) = [\psi_1, \psi_2, \dots, \psi_8]$$

Ecuación 2.6 Preprocesamiento de la señal EMG desconocida $W(N)$

Para las señales EMG de la base de datos, B , se aplica el mismo procesamiento: rectificación y filtrado. Esto se muestra en la Ecuación 2.7.

$$z_{i(m)} = \mathcal{F}(abs(b_{i(m)}), \omega_c)$$

Ecuación 2.7 Preprocesamiento de las señales EMG de la base de datos

Donde: $z_{i(m)}$ es la señal procesada correspondiente al canal i de la repetición m

$b_{i(m)} \in B$ es la señal EMG de la base de datos correspondiente al canal i de la repetición m (Ecuación 2.2).

El conjunto de todas las señales $z_{i(m)}$ preprocesadas del entrenamiento se define en la Ecuación 2.8 como Z .

$$Z = \begin{bmatrix} z_{1(1)} & \cdots & z_{8(1)} \\ \vdots & \ddots & \vdots \\ z_{1(30)} & \cdots & z_{8(30)} \end{bmatrix}$$

Ecuación 2.8 Preprocesamiento de las señales de la base de datos B

El filtro de Butterworth, \mathcal{F} , tiene como parámetros el orden del filtro y la frecuencia de corte ω_c . Puesto que el modelo matemático utilizado en este proyecto considera que la actividad muscular corresponde a la gaussiana modulante (sección 0), se utiliza un filtro pasa bajos que reduzca las componentes de alta frecuencia. En la Figura 2.8 se muestra la señal EMG pura, W , y la señal rectificada, $abs(W)$, para cada canal. En la Figura 2.9 se presenta la familia de curvas para diferentes frecuencias de corte ω_c .

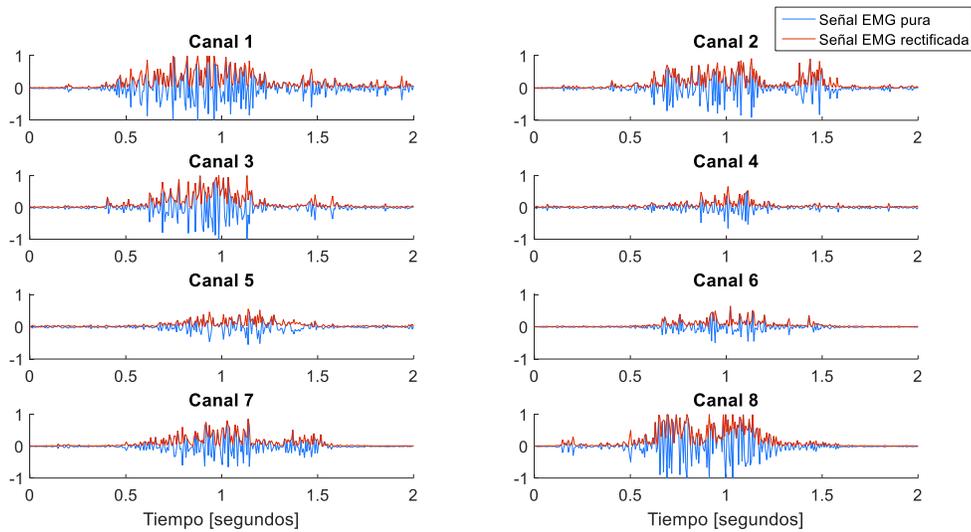


Figura 2.8 Señal EMG pura y señal EMG rectificada de los ocho canales

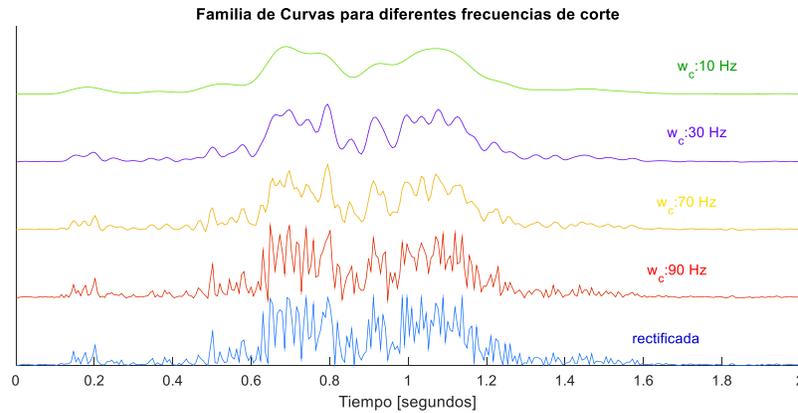


Figura 2.9 Familia de Curvas para diferentes frecuencias de corte ω_c

Se busca que la señal filtrada se asemeje lo más posible a la señal correspondiente a la actividad muscular. El criterio para determinar la frecuencia de corte, ω_c , es seleccionar la frecuencia a la cual se obtenga un mejor desempeño de clasificación. En este trabajo se observó que el cálculo de las distancias DTW presenta mejores resultados al utilizarse filtros pasa bajos con frecuencias de corte muy bajas. Los parámetros escogidos para el preprocesamiento de la señal son los siguientes:

Tabla 2.4 Parámetros del filtrado de la señal EMG

Parámetro	
Filtro	<i>Butterworth</i>
Tipo	<i>Pasa bajos</i>
Orden	<i>Cuarto orden</i>
Frecuencia de corte ω_c	$10\pi \left[\frac{rad}{seg} \right]$

Al aplicar el procesamiento (rectificado y filtrado) de las señales EMG sobre cada gesto se encuentran notables similitudes. En la Figura 2.10 se presenta las señales EMG procesadas, ψ , para el gesto *puño (Fist)*. Se grabó la señal EMG durante 10 segundos, en los cuales se realizaron 7 repeticiones consecutivas del gesto. Visualmente se puede notar que las señales presentan gran similitud entre repeticiones. Para facilidad de observación, se presenta las curvas de dos canales EMG en cada eje. En el ANEXO I se presenta las figuras de todos los gestos rectificadas y filtradas.

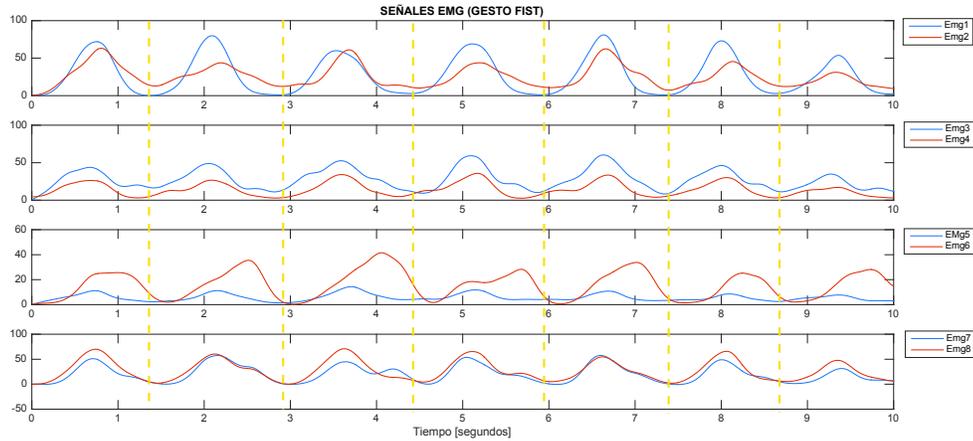


Figura 2.10 Señal EMG después del preprocesamiento

Comparación de las señales electromiográficas utilizando DTW

El sistema de clasificación de gestos requiere comparar la similitud de las señales electromiográficas correspondientes a cada gesto. Para las señales EMG que transcurren en el tiempo y que tienen diferente duración entre gestos, no se puede utilizar la distancia euclidiana tradicional que requiere que las señales estén sincronizadas y que tengan la misma longitud. En el presente estudio técnico se utiliza el algoritmo DTW para calcular la similitud de entre las señales electromiográficas. En la Ecuación 2.9. se calcula la distancia, d , entre la señal EMG procesada, Ψ , y las señales EMG de la base de datos, Z .

$$d_{i(m)} = DTW(\psi_i, z_{i(m)})$$

Ecuación 2.9 Distancias DTW de la señal EMG por canal

Donde: $DTW(a, b)$ es el cálculo de la distancia DTW entre la señal a y b , (el cálculo del algoritmo DTW se explica en la siguiente sección)

ψ_i se definió en la Ecuación 2.5

$z_{i(m)}$ se definió en la Ecuación 2.7.

Descripción del algoritmo de alineamiento temporal dinámico (DTW)

El algoritmo de alineamiento temporal dinámico (DTW, por sus siglas en inglés, *Dynamic Time Warping*) es una técnica para encontrar un alineamiento óptimo entre dos series de tiempo dadas [40] [41]. Un alineamiento que utilice la distancia euclidiana relacionará el i ésimo punto de la señal A con el i ésimo punto de la señal B dando pobres resultados al

tenerse diferentes longitudes de las señales o ante desplazamientos de fase [42]. El alineamiento del algoritmo DTW se realiza encontrando la correspondencia de cada uno de los puntos entre ambas series que dé como resultado la menor distancia. En la Figura 2.11 se compara el alineamiento de dos series de tiempo al utilizar la distancia euclidiana y al utilizar DTW.

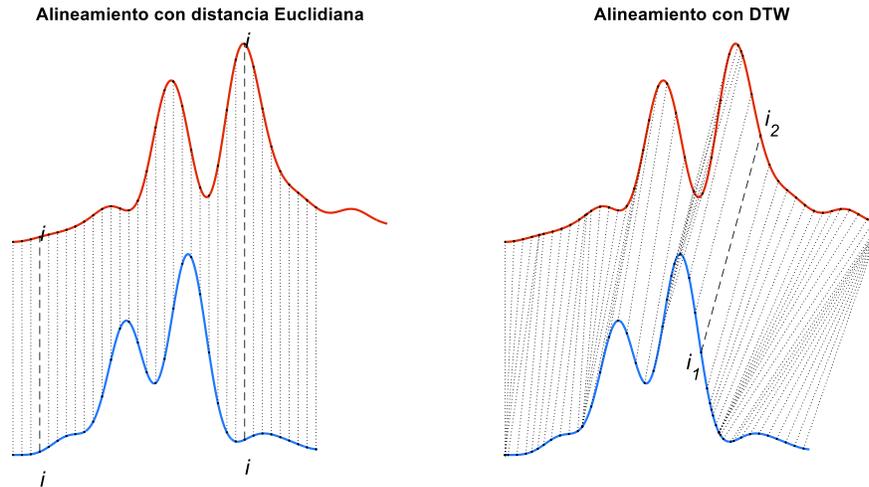


Figura 2.11 Alineamiento de dos señales con (a) distancia euclidiana y (b) DTW

El cálculo del DTW requiere encontrar una ruta dentro de la matriz de distancias, que minimice la distancia entre las dos señales de tiempo [43]. La Figura 2.12 contiene un ejemplo sobre la matriz de distancias y la ruta óptima para las dos secuencias A y B . En esta figura, cada casilla de la matriz de distancias corresponde a la distancia entre un punto i de la secuencia A con un punto j de la secuencia B . Se resalta la ruta óptima para alinear las secuencias A y B .

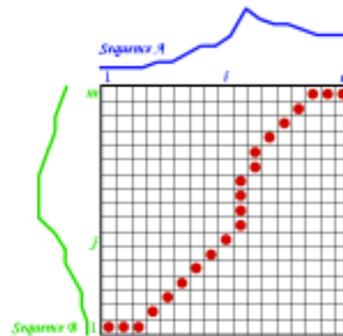


Figura 2.12 Matriz de distancias y ruta óptima en el cálculo del DTW [42]

Matemáticamente, considérese las series de tiempo A y B , de longitud l_a y l_b , respectivamente tal que:

$$A = a_1, a_2, a_3 \dots a_{l_a}$$

$$B = b_1, b_2, b_3 \dots b_{l_b}$$

La matriz de distancias, o matriz de costos, C está formada por los elementos $c(i, j)$ tal que:

$$c(i, j) = (a_i - b_j)^2$$

Ecuación 2.10 Cálculo de los elementos en la Matriz de Costos C

Para encontrar la mejor alineación ente las dos series, A y B , se busca una ruta a través de la matriz C que minimice la distancia acumulativa entre ellas [44]. Esta ruta, R , se define como el conjunto de puntos, $r_q = (i, j)_q$, tal que:

$$R = r_1, r_2, r_3 \dots r_q \dots r_Q, \quad r_q = (i, j)_q, \quad \max(l_a, l_b) \leq Q \leq l_a + l_b - 1$$

Además, la ruta óptima, R , es aquella ruta que minimiza la función de costo:

$$DTW(A, B) = \min \sqrt{\sum_{q=1}^Q r_q}$$

Ecuación 2.11 Cálculo del algoritmo DTW a partir de la ruta óptima

Esta ruta óptima, además, puede encontrarse utilizando programación dinámica con la distancia acumulativa $\gamma(i, j)$:

$$\gamma(i, j) = d(i, j) + \min\{\gamma(i - 1, j - 1), \gamma(i - 1, j), \gamma(i, j - 1)\}$$

Ecuación 2.12 Distancia acumulativa mínima

Donde: $\gamma(1, 1) = d(1, 1)$

$\gamma = \infty$, cuando $i = 0$ o $j = 0$.

Nótese que al utilizar la distancia acumulativa se reduce el tiempo de procesamiento, pues no se calculan todos los puntos de la matriz de costos, sino solo aquellos de las celdas adyacentes, reduciendo significativamente el número de puntos a considerar. En la Figura 2.13 se muestra el concepto de las celdas adyacentes en el cálculo de la distancia acumulativa. En el presente trabajo, el cálculo del algoritmo DTW, se realiza utilizando la distancia acumulativa, Ecuación 2.13.

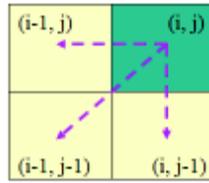


Figura 2.13 Celdas adyacentes para el cálculo de la distancia acumulativa γ [44]

$$DTW(A, B) = \gamma(l_a, l_b)$$

Ecuación 2.13 Cálculo del algoritmo DTW a partir de la distancia acumulativa

Debido a que el número de posibles rutas en la matriz de costos crece exponencialmente con el tamaño del vector A y B , se aplican restricciones al calcular la distancia DTW [42]. Estas restricciones previenen algún alineamiento inapropiado, además de reducir el número de rutas consideradas durante el proceso de alineamiento [44]. En la Figura 2.14 se muestran posibles rutas de alineamiento entre las secuencias A y B .

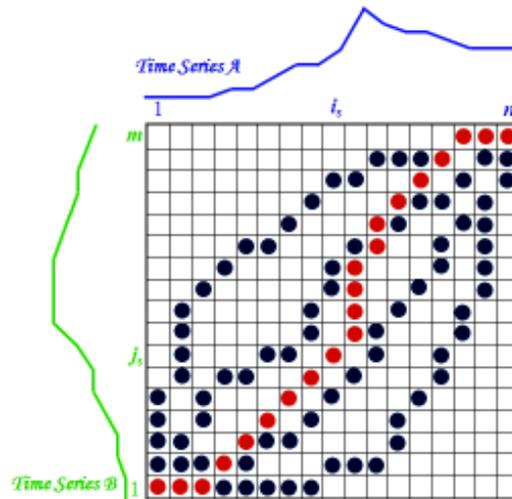


Figura 2.14 Posibles rutas en la matriz de costos [42]

Se aplican 4 restricciones en el cálculo del algoritmo DTW: la restricción de frontera, la restricción de continuidad y la restricción de monotonía, la restricción de ventana. A continuación, se describen cada una de estas restricciones.

Restricción de Frontera: La ruta del alineamiento inicia en el primer punto y termina en el último de ambas señales, tal que $r_1 = (1,1)$, $r_Q = (l_a, l_b)$. Con esta condición se asegura que el alineamiento no considere parcialmente a una de las secuencias [42]. En la Figura 2.15 se presenta una matriz de costos donde se incluye dos rutas de alineamiento: una correcta y una que no cumple con la restricción de frontera. La restricción de Frontera se expresa como:

$$i_1 = 1, i_Q = l_a, \quad j_1 = 1, j_Q = l_b$$

Ecuación 2.14 Restricción de frontera

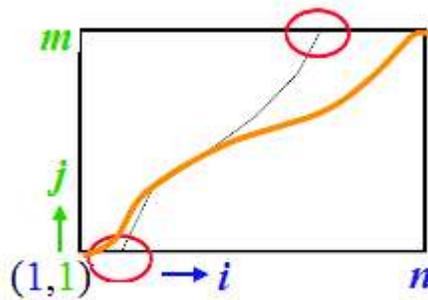


Figura 2.15 Restricción de Frontera en la matriz de costos [42]

Restricción de monotonía: La ruta del alineamiento no retrocede en el tiempo [42]. Esta condición se cumple al realizar el cálculo de la distancia acumulativa en la Ecuación 2.12 (ver Figura 2.13). Pues, la distancia acumulativa, para un punto (i, j) solo considera los puntos adyacentes $(i - 1, j - 1)$, $(i - 1, j)$, $(i, j - 1)$. Esta condición garantiza que ciertas características de la señal no se repitan en el alineamiento. En la Figura 2.16 se presenta una matriz de costos con dos rutas de alineamiento donde una de ellas no cumple la restricción de monotonía. Por lo que la restricción de monotonía se puede expresar como:

$$i_{q-1} \leq i_q, \quad j_{q-1} \leq j_q$$

Ecuación 2.15 Restricción de monotonía

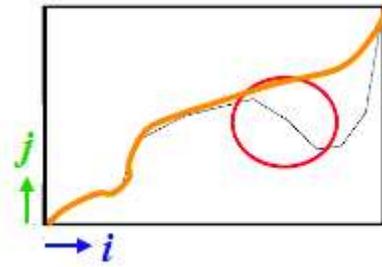


Figura 2.16 Restricción de Monotonía [42]

Restricción de continuidad: La ruta del alineamiento no salta en el dominio del tiempo. Con esta condición se garantiza que el alineamiento no omita características importantes de ninguna de las dos señales [42]. Esta restricción se cumple en el cálculo del algoritmo DTW al considerar incrementos unitarios en los índices i, j de la distancia acumulativa, Ecuación 2.16.

$$i_q - i_{q-1} \leq 1, \quad j_q - j_{q-1} \leq 1$$

Ecuación 2.16 Restricción de continuidad

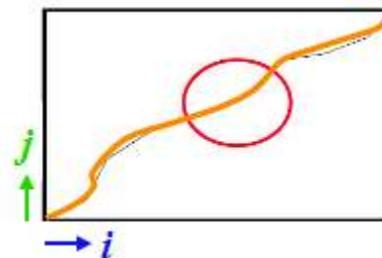


Figura 2.17 Restricción de Continuidad [42]

Estas tres restricciones son aplicadas al realizar el cálculo de la distancia acumulativa. En la Figura 2.18 se incluye un ejemplo de una ruta de alineamiento idónea, junto con una ruta que incumple cada una de las restricciones anteriores.

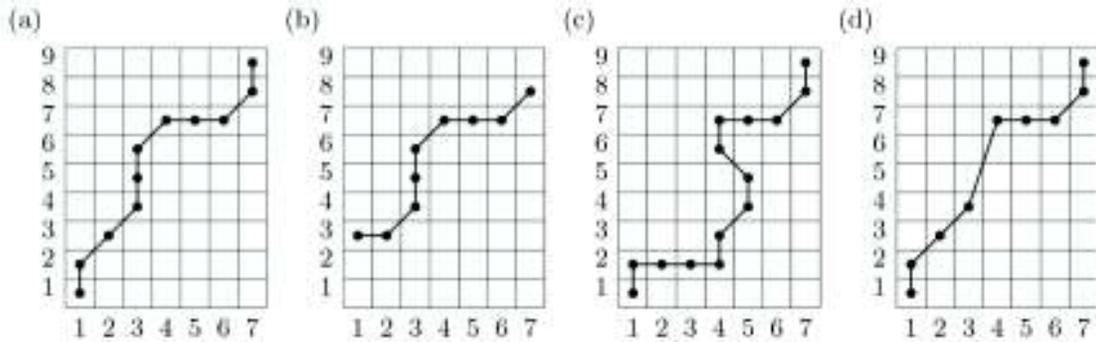


Figura 2.18 Restricciones del cálculo DTW: a) ruta de alineamiento idónea, b) ruta que incumple la restricción de frontera, c) ruta que incumple la restricción de monotonía, y d) ruta que incumple la restricción de continuidad [41]

Restricción de ajuste de ventana: Algunos autores incluyen una restricción adicional. Esta restricción limita los puntos de las esquinas considerados en el alineamiento. Se incluye esta condición puesto que un alineamiento óptimo es improbable que se encuentre demasiado alejado de la diagonal de la matriz de costos [42]. Para ello, se considera una distancia fija mínima de diferencia entre i y j . En la Figura 2.19 se muestran ambos tipos de restricción de ventana.

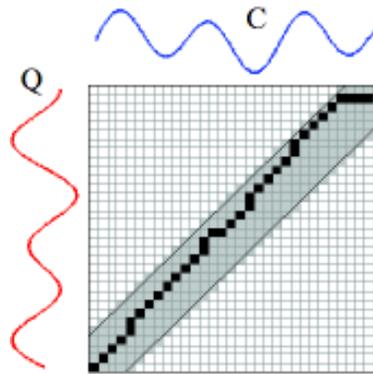


Figura 2.19 Restricción de ventana [44]

Cálculo de la distancia DTW entre las señales EMG

El algoritmo DTW retorna una distancia por cada canal del Myo Armband (Ecuación 2.9). Esto resulta en ocho distancias entre el gesto desconocido y un gesto de la base de datos. En la Ecuación 2.17 se define una distancia equivalente, D , a partir de las distancias d_i .

$$D_m = \sum_{i=1}^8 d_{i(m)}$$

Ecuación 2.17 Distancia DTW total

Donde: D_m es la distancia total entre la señal desconocida, $W(N)$, y la señal EMG de la base de datos de la repetición m

$d_{i(m)}$ se definió en la Ecuación 2.9

El conjunto de distancias, D_m , con $m = 1, \dots, 30$, representa las distancias entre la señal EMG desconocida y cada una de las señales EMG de la base de datos, Ecuación 2.18. Este conjunto de distancias, θ , es utilizado por el algoritmo kNN para la predicción de la clasificación, $y(N)$.

$$\theta = [D_1, D_2, \dots, D_{30}]$$

Ecuación 2.18 Conjunto de distancias entre la señal EMG desconocida y todas las señales del entrenamiento

Donde: D se definió en la Ecuación 2.17

Clasificación de las señales electromiográficas utilizando kNN

A la señal electromiográfica que transcurre en el tiempo, $W(N)$, el sistema de clasificación de gestos asigna con uno de los gestos de la rutina de entrenamiento. Esta clasificación de la señal EMG desconocida se realiza utilizando el algoritmo kNN.

Descripción del clasificador de los k vecinos más cercanos (kNN)

El clasificador k vecinos más cercanos kNN (*k Nearest Neighbors* por sus siglas en inglés) es un método no paramétrico de Aprendizaje Computacional supervisado. Un método no paramétrico, en estadística, es un método que no asume ninguna función de distribución para los datos en análisis [45] [46]. El clasificador kNN memoriza todos los casos posibles del entrenamiento, pues no hay un modelo asociado al concepto aprendido. Debido a esto, el costo computacional del kNN es cero en el entrenamiento; pero recae mayormente en la predicción [47].

El clasificador kNN asume que cada punto de análisis (señales del entrenamiento y señales desconocidas) está en un espacio métrico de características [48]. Por lo que el algoritmo necesita de una función de distancia [47] para comparar la similitud de las señales del conjunto de datos de entrenamiento con la señal desconocida. Existe varias funciones de

distancia posibles. Para valores continuos está la distancia Euclidiana, distancia Manhattan y distancia de Minkowski. En cambio, para valores categóricos, se utiliza la distancia de Hamming [49]. En el presente estudio técnico se utiliza el algoritmo DTW como función de distancia entre las señales EMG.

La predicción del clasificador kNN se basa en una votación entre los K vecinos más cercanos al punto de evaluación. Para ello, un entero positivo K es especificado [46]. Es conveniente establecer K como un número impar, para de esta manera evitar empates en la votación [50]. En la Figura 2.20 se representa el cambio en la predicción del clasificador, para un ejemplo de dos clases, al elegir diferentes valores de K .

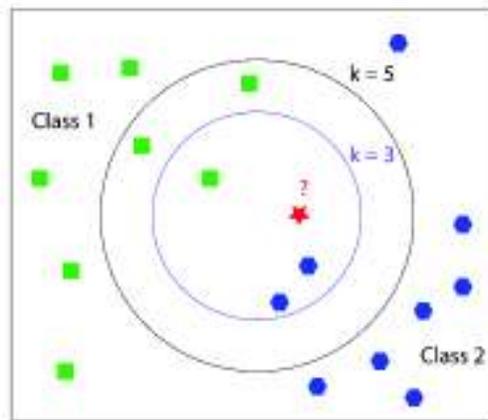


Figura 2.20 Clasificador kNN entre dos clases para $K = 3$ y $K = 5$ [46]

La predicción del clasificador kNN depende del número de vecinos K tomados en cuenta en la votación. Se Escoge el valor de K buscando reducir el error que se produce en el entrenamiento. Para valores de K muy grandes, se produce un borde mucho más suavizado, lo que puede reducir el ruido de las etiquetas de clase; sin embargo, cuando K es igual al número de muestras del entrenamiento, $K = N$, el sistema predice únicamente a la clase mayoritaria [51].

El hecho de que un algoritmo de aprendizaje tienda a una solución óptima al incrementarse el número de muestras del entrenamiento N se conoce como consistencia. Un algoritmo de aprendizaje se considera universalmente consistente si este es consistente para todas las distribuciones de probabilidad [51].

En el clasificador kNN, la consistencia universal se da si:

$$\text{si } K \rightarrow \infty, \quad \text{entonces } \frac{K}{N} \rightarrow 0, \quad \text{cuando } N \rightarrow \infty$$

Ecuación 2.19 criterio de consistencia universal [50]

Donde: K es el número de vecinos más cercanos considerados

N es el número de muestras del entrenamiento

En la práctica, N es finito, por lo que K no puede ser mayor a N . En ese caso, se escoge un valor de K que resulte en un crecimiento lento en comparación a N [51]. Para ello se el parámetro K se establece con:

$$K = \log_2(N)$$

Ecuación 2.20 Valor del parámetro K en el clasificador kNN

En el presente estudio técnico, el número de muestras de entrenamiento es 30, cada gesto fue repetido 5 veces. Utilizando la Ecuación 2.20, el valor de K resultante es 4.9. Al redondear esta cifra se consigue el número de vecinos más cercanos para el sistema de clasificación de gestos ($K = 5$).

Clasificación utilizando kNN

Para la clasificación, se consideran los 5 vecinos más cercanos a la señal EMG desconocida, $W(N)$. Estos vecinos más cercanos corresponden a las señales de la base de datos Z con menor distancia DTW. Por lo que, el vector de distancias, D_m (ver Ecuación 2.17), es ordenado para conseguir los 5 menores elementos, con su etiqueta correspondiente, $L(x_i)$.

$$X = \underset{K}{\operatorname{argmin}}(\theta)$$

Ecuación 2.21 Clasificador kNN: ordenamiento

Donde: $\underset{K}{\operatorname{argmin}}(a)$ es el operador que calcula los K menores elementos de a

X es un vector de K elementos formado por x_i

x_i corresponde al i ésimo gesto dentro de los K vecinos más cercanos a la señal desconocida $W(N)$.

θ se definió en la Ecuación 2.18.

K es el número de vecinos y se estableció en 5 en la Ecuación 2.20.

Cada gesto x_j , dentro de los vecinos más cercanos, X , tiene asignada una probabilidad $P_{(x_j)}$. Esta probabilidad depende del número de votos, $\#x_j$, correspondientes al gesto x_j . La probabilidad viene dada por la Ecuación 2.22:

$$P_{(x_j)} = \frac{\#x_j}{K}$$

Ecuación 2.22 Probabilidad de cada gesto usando kNN

Donde: $P_{(x_j)}$ es la probabilidad del gesto x_i , con $j = 1, 2, \dots, 6$.

$\#x_j$ es el número de elementos, que corresponden a x_j en el vector de K vecinos, X .

El algoritmo kNN, en el sistema de clasificación de gestos, considera un umbral de probabilidad para la predicción. La predicción en la iteración N , $y(N)$, corresponde al gesto, x_j , más votado (mayor probabilidad P_x) entre los vecinos más cercanos solo cuando el número de votos, $\#x_j$, supera el umbral, τ . Caso contrario, el sistema retorna el gesto **posición de reposo**, x_0 , (sección 0). Esto se expresa en la Ecuación 2.23.

$$y_n = \begin{cases} x_j, & P_{(x_j)} \geq \tau \\ x_0, & P_{(x_j)} < \tau \end{cases}$$

Ecuación 2.23 Predicción del sistema de clasificación de gestos

Donde: $y(N)$ es la predicción de la iteración N

x_j es el gesto con mayor probabilidad $P_{(x_j)}$

x_0 es el gesto posición de reposo

$P_{(x_j)}$ es la probabilidad del gesto x_j

τ es el umbral de probabilidad

En el presente estudio técnico se consideran 5 vecinos para la votación. Se escogió este número de vecinos considerando la condición de consistencia de la Ecuación 2.20. Entre los vecinos deben existir 4 o más pertenecientes a la misma categoría, (umbral de 80%). Caso contrario, la predicción asume el estado de reposo. En la Tabla 2.5 se resume los parámetros del clasificador kNN.

Tabla 2.5 Parámetros del clasificador kNN

Parámetro	Valor	Unidad
<i>K Vecinos</i>	5	[unidades]
<i>umbral probabilidad, τ</i>	80	%

Etapa de posfiltrado de la seña

El sistema de clasificaci3n de gestos retorna una clasificaci3n (predicci3n) cada 200 ms. Esta seña resultado, $y(N)$, puede tener discontinuidades que afecten el funcionamiento del brazo rob3tico de 3 GDL. Para ello, se aplica un filtro a la salida. Este filtro consiste en asignar a $Y(N)$ el gesto **posici3n de reposo** (definido en la secci3n 0) cuando la clasificaci3n actual y anterior no coinciden.

$$Y(N) = \begin{cases} y(N), & y(N) \neq y(N - 1) \\ x_0, & y(N) = y(N - 1) \end{cases}$$

En la Figura 2.21 se muestra el resultado de la predicci3n $y(N)$, junto con la seña filtrada $Y(N)$.

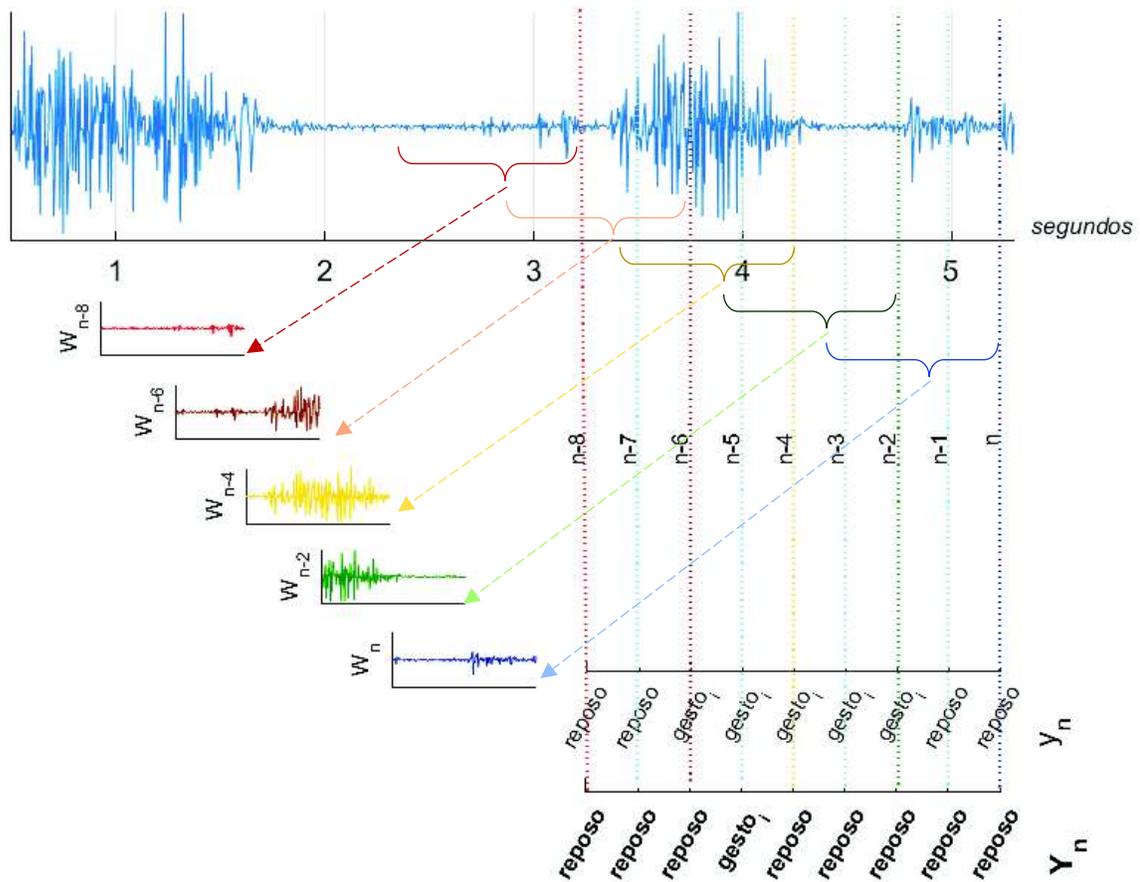


Figura 2.21 Pos-procesamiento de la señal

La señal $Y(N)$ es la respuesta final del sistema de clasificación de gestos que se utiliza como comandos para el brazo robótico de 3 grados de libertad. Esta señal se puede considerar como un único pulso que tiene el valor "reposo" por defecto. Se implementa, a continuación, el mando a distancia de un brazo robótico de 3 GDL a partir de la señal $Y(N)$.

2.4 Mando a distancia del brazo robótico de 3 GDL

En el presente apartado se describe el mando a distancia del brazo robótico de 3 GDL. El brazo robótico es implementado con el kit Lego Mindstorms EV3. Las señales de mando son enviadas vía Bluetooth desde el computador hacia el ladrillo inteligente EV3. Las señales de referencia son los ángulos de Tait-Bryan (alabeo, cabeceo y guiñada), calculados a partir de la medición de la orientación del Myo Armband. Se implementa un control de posición angular PID para cada grado de libertad del brazo robótico.

Configuración del brazo robótico

El instituto Americano de Robótica define a un robot como “*Un manipulador reprogramable, multifuncional, designado para mover materiales, partes, herramientas o dispositivos especializados a través de varios movimientos programados que cumplen diferentes tareas*” [52]. Un manipulador es una secuencia de eslabones unidos en pares por articulaciones. Cada uno de los movimientos independientes que puede realizar cada articulación con respecto a la anterior se denomina grado de libertad [53]. La clasificación de los robots manipuladores, según la Asociación Francesa de Robótica Industrial (AFRI) [53] se presenta en la Tabla 2.6.

Tabla 2.6 Clasificación de los robots según la AFRI [53]

Tipo	Descripción
Tipo A	Manipulador con control manual o mando a distancia
Tipo B	Manipulador automático con ciclos preajustados; regulación mediante fines de carrera o topes, control por PLC, accionamiento neumático, eléctrico o hidráulico
Tipo C	Robot programable con trayectoria continua o punto a punto. Carece de conocimientos sobre su entorno.
Tipo D	Robot capaz de adquirir datos de su entorno, y readaptar su tarea.

Bajo la clasificación de la AFRI, el robot de 3 grados de libertad implementado en este trabajo pertenece al Tipo A. La configuración escogida del brazo robótico se muestra en el esquema de la Figura 2.22, donde el eje de giro es el eje Z_i .

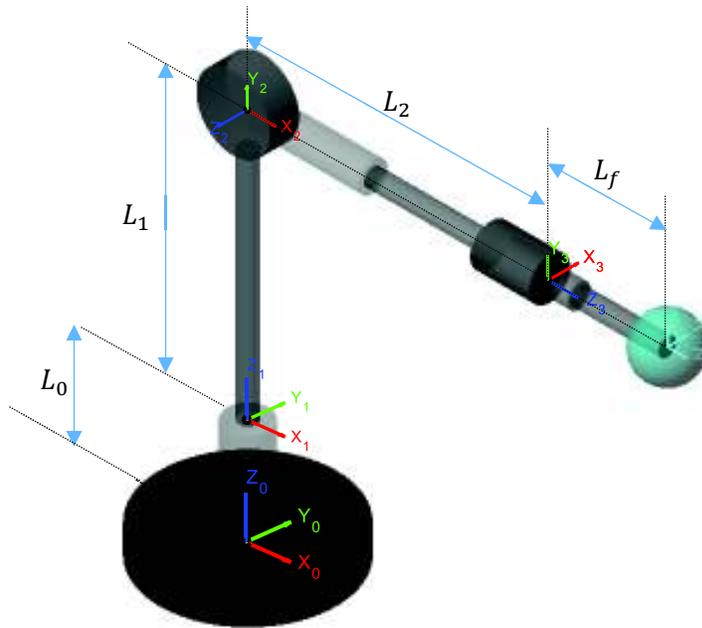


Figura 2.22 Configuración del brazo robótico

Kit Lego Mindstorms EV3

Lego Mindstorms es una serie de kits robóticos educativos creada por la compañía LEGO®. La serie Lego Mindstorms se caracteriza por poseer sensores y actuadores integrados bajo una misma plataforma, lo que lo convierte en un kit de robótica de fácil uso e implementación [54]. Lego Mindstorms EV3 es último kit de la serie. En la Figura 2.23, se muestran los componentes del Lego Mindstorms EV3.



Figura 2.23 Lego Mindstorms EV3 [54]

Lego Mindstorms EV3 se caracteriza por poseer una CPU, sensores, actuadores; elementos de construcción y cables. La CPU del Lego Mindstorms EV3 se conoce como ladrillo

Inteligente (*brick EV3*). Este está formado por un microcontrolador ARM9 de 32 bits programable. Posee 4 puertos de entrada, 4 puertos de salida, permite la comunicación a través de Wifi, Bluetooth y USB. Además, tiene una pantalla gráfica LCD 178x128 pixeles, interfaz con 6 botones, y puerto para memoria miniSDHC, entre otros [55].



Figura 2.24 Ladrillo inteligente EV3 [54]

El kit Lego Mindstorms EV3 posee dos tipos de actuadores: servomotor largo *M* (*EV3 Large Servo Motor 45502*), y servomotor mediano *m* (*EV3 Medium Servo Motor 45503*). Estos dos tipos de motores son alimentados a 9v; se diferencian en su torque y velocidad nominales [56]. En la Figura 2.25 se muestra ambos motores. Las características de los motores se presentan en la Tabla 2.7.2



Figura 2.25 Actuadores de Lego Mindstorms EV3 a) motor Largo, b) motor mediano [54]

Tabla 2.7 Características de los motores del EV3 [56]

Motor	Largo	Mediano
Velocidad	160 – 170 RPM	240 – 250 RPM
Torque Nominal	20 N.cm	8 N.cm
Peso	82 g	39 g
Corriente en vacío	60 mA	80 mA
Torque a rotor bloqueado	43 N.cm	15 N.cm

Implementación del brazo robótico de 3 GDL

En el presente estudio técnico se utiliza el kit Lego Mindstorms para la implementación del brazo robótico. La configuración de manipulador escogida (Figura 2.22), utiliza tres motores largos para el giro de cada grado de libertad, y adicionalmente, se utiliza un motor mediano para el movimiento de una pinza robótica.

El brazo robótico se diseñó considerando una fácil separación de sus partes. Por lo que su construcción se puede dividir en: construcción de la base L_0 , construcción del primer eslabón L_1 , construcción del segundo eslabón L_2 , construcción del efector final L_f (pinza robótica).

Todas las articulaciones están conectadas a los motores del EV3 a través de una serie de engranes. La ventaja de utilizar engranes es que aumentan el torque y reducen la velocidad del eje. La relación de engranes se calcula considerando que la potencia a ambos lados idealmente se mantiene constante. Se presenta en la Ecuación 2.24 la relación de torques y velocidades. Una desventaja de usar engranes, sin embargo, es que aumentan la oscilación del eje de giro, lo que produce oscilaciones y errores de posición [55]. La relación torque al lado fijo y al lado móvil es proporcional al número de dientes de los engranes.

$$\frac{T_f}{T_m} = \frac{\omega_m}{\omega_f} = \frac{Z_m}{Z_f}$$

Ecuación 2.24 Velocidad y torque a partir de la relación de engranes

Donde: T es el torque
 ω es la velocidad angular
 Z el número de dientes del engrane
 m es el lado seguidor del engrane
 f es el lado principal del engrane

El diseño de cada parte del brazo robótico depende en gran medida de los engranes que se requiera. El conjunto de engranes disponibles en el kit Lego se presenta en la Figura 2.26. Utilizando esta gráfica, que contiene el número de dientes de cada tipo de engrane, se realizó el diseño de cada parte del brazo robótico.

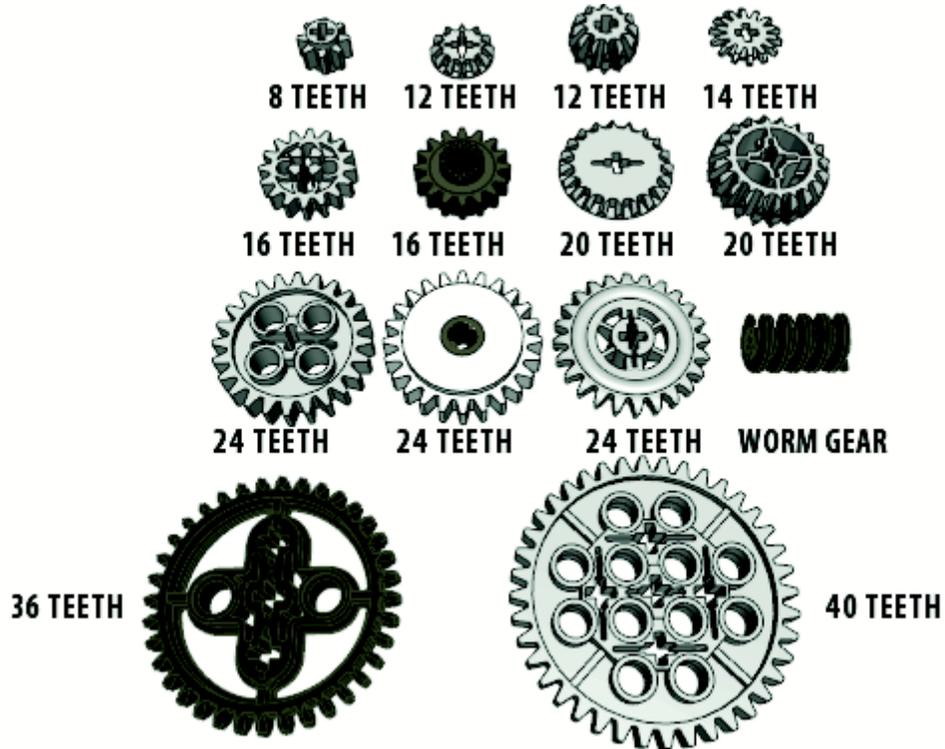


Figura 2.26 Tipos de engranes Lego [55]

El modelo de brazo robótico implementado fue modelado utilizando el software Lego Digital Designer. Este software, abreviado LDD, sirve para el diseño de proyectos en Lego; este contiene la enumeración de piezas, permite la construcción de modelos digitalmente, y ofrece instrucciones detalladas para su reconstrucción. En la Figura 2.27 se muestra la interfaz de usuario de Lego Digital Designer. En el ANEXO II se encuentra las instrucciones detalladas para la reconstrucción del brazo robótico implementado generadas por el software Lego Digital Designer.



Figura 2.27 Interfaz de usuario del software Lego Digital Designer

Construcción de la base

La base debe ser capaz de soportar el peso del brazo robótico. La base contiene, además, al primer motor largo, correspondiente al ángulo de guiñada (*yaw*). Para evitar que se deslice en la superficie, la base se ajusta con la mesa de trabajo.

El motor largo, M_1 , es responsable del movimiento del primer grado de libertad Q_1 alrededor de del eje Z_0 . El motor Largo M_1 se conecta a una serie de engranes para aumentar el torque y disminuir la velocidad. En la Figura 2.28 se muestra la configuración de engranes para tener un aumento de 6.75 veces el torque del motor.



Figura 2.28 Relación de engranes para el primer grado de libertad Q_1

$$\text{Relación de engranes}_{Q_1} = \frac{36}{12} * \frac{36}{16} = 6.75$$

Ecuación 2.25 Relación de engranes de Q_1

En la Figura 2.29 se presenta la base del brazo robótico que contiene la articulación Q_1 . En el ANEXO II se encuentran las instrucciones para su construcción. El primer eslabón se conecta en el eje de giro marcado. El motor largo M_1 debe conectarse vía USB al puerto A del ladrillo inteligente EV3.

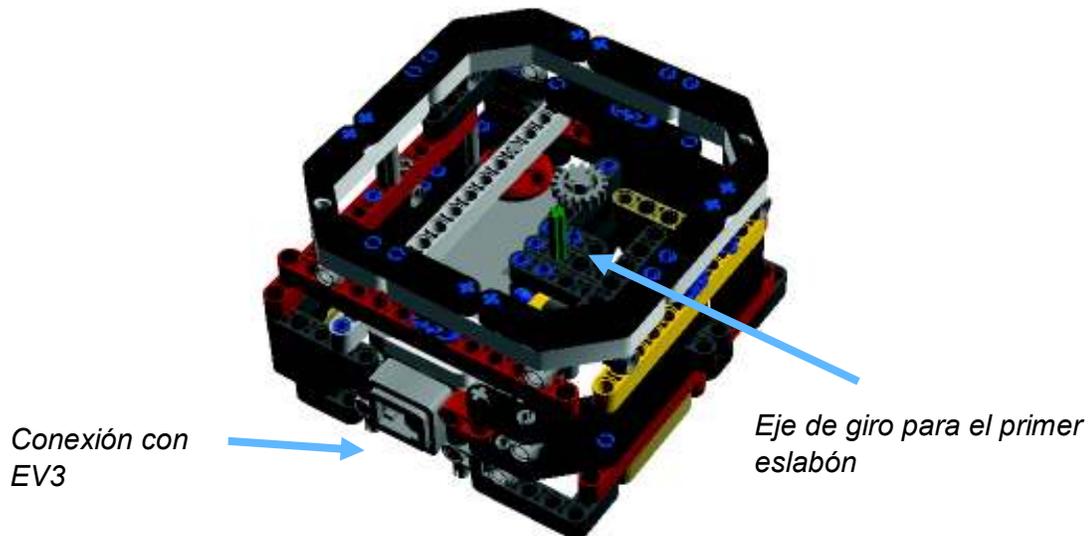


Figura 2.29 Base del brazo robótico

Construcción del primer eslabón

El primer eslabón se monta sobre la base descrita en la sección anterior. En este eslabón se utiliza el segundo motor Largo M_2 . Para evitar que el movimiento enrede y desconecte los cables, se monta al ladrillo inteligente EV3 en este eslabón. De esta manera se reduce sustancialmente el giro de los cables alrededor del robot. El ubicar al ladrillo EV3 en este eslabón consigue, además, un contrapeso para el balance del brazo robótico entero. La articulación conectada a este eslabón es Q_2 . Para el giro de Q_2 se utiliza una serie de cinco engranes.

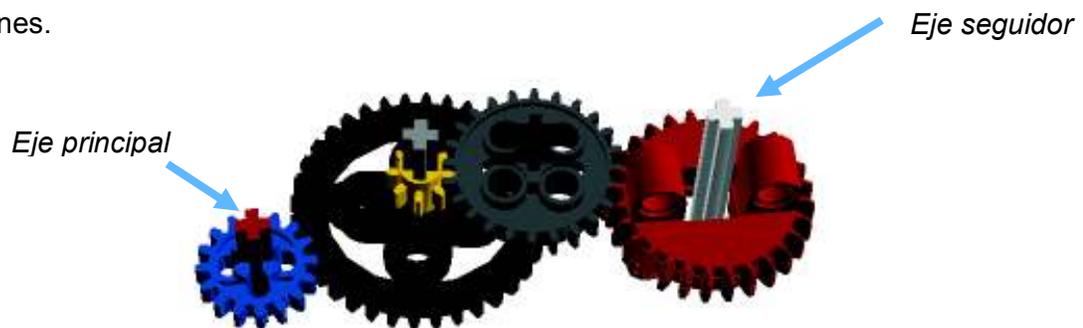


Figura 2.30 Relación de engranes para Q_2

$$\text{Relación de engranes}_{Q_1} = \frac{36}{16} * \frac{24}{8} * \frac{28}{24} = 7.875$$

Ecuación 2.26 Relación de engranes para Q_2

En la Figura 2.31 muestra el modelo en Lego Digital Designer del primer eslabón que contiene el segundo grado de libertad Q_2 . Las instrucciones detalladas para la construcción se incluyen en el ANEXO II. En este eslabón se encuentra el ladrillo inteligente al que se deben conectar los cuatro motores. Además, este eslabón se conecta con la base del brazo robótico y el segundo eslabón.

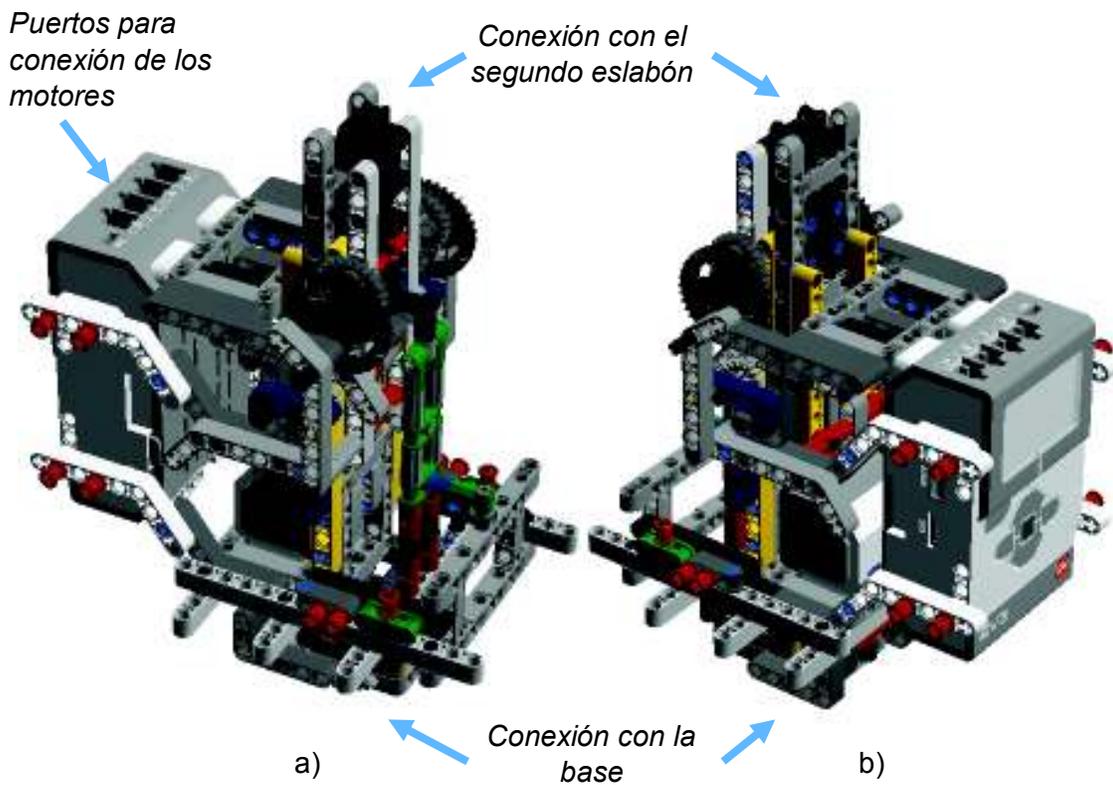


Figura 2.31 Primer Eslabón del brazo robótico a) vista frontal, b) vista trasera

Construcción del segundo eslabón

El segundo eslabón del brazo robótico se mueve alrededor del eje Z_2 , (ver sección 0) utilizando el tercer motor largo M_3 . Además, posee la articulación Q_3 , ángulo de alabeo (*roll*), que sostiene el efector final. Este eslabón se diseñó considerando la menor longitud posi-

ble, pues mientras más largo es el eslabón, mayor el torque que se requiere en la articulación Q_2 . En el caso de este eslabón, se utiliza una serie de engranes que cambian el eje de giro con respecto al motor. Para instrucciones detalladas, ver ANEXO II.

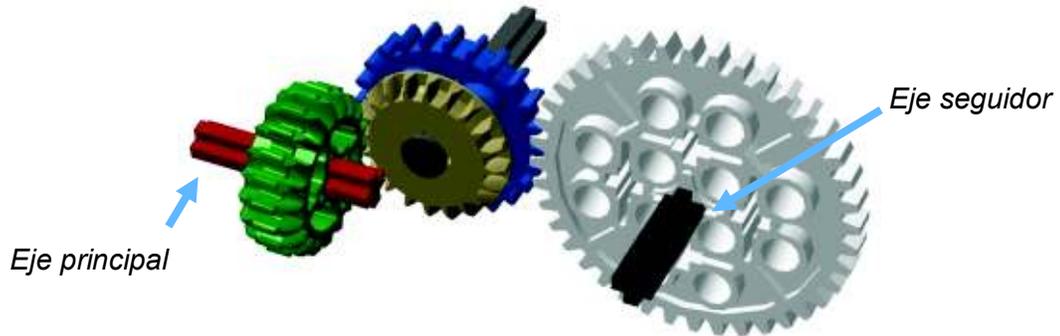


Figura 2.32 Serie de engranes para el tercer grado de libertad Q_3

$$\text{Relación de engranes}_{Q_1} = \frac{20}{20} * \frac{40}{24} = 1.667$$

Ecuación 2.27 Relación de engranes Q_3

En este caso, la relación de engranes no varía en gran medida el torque, su finalidad es cambiar el eje de giro del motor para que concuerde con el diseñado para Q_3 . El motor M_3 se debe conectar al puerto C del labrido Ilo inteligente. El segundo eslabón se muestra en la Figura 2.33.

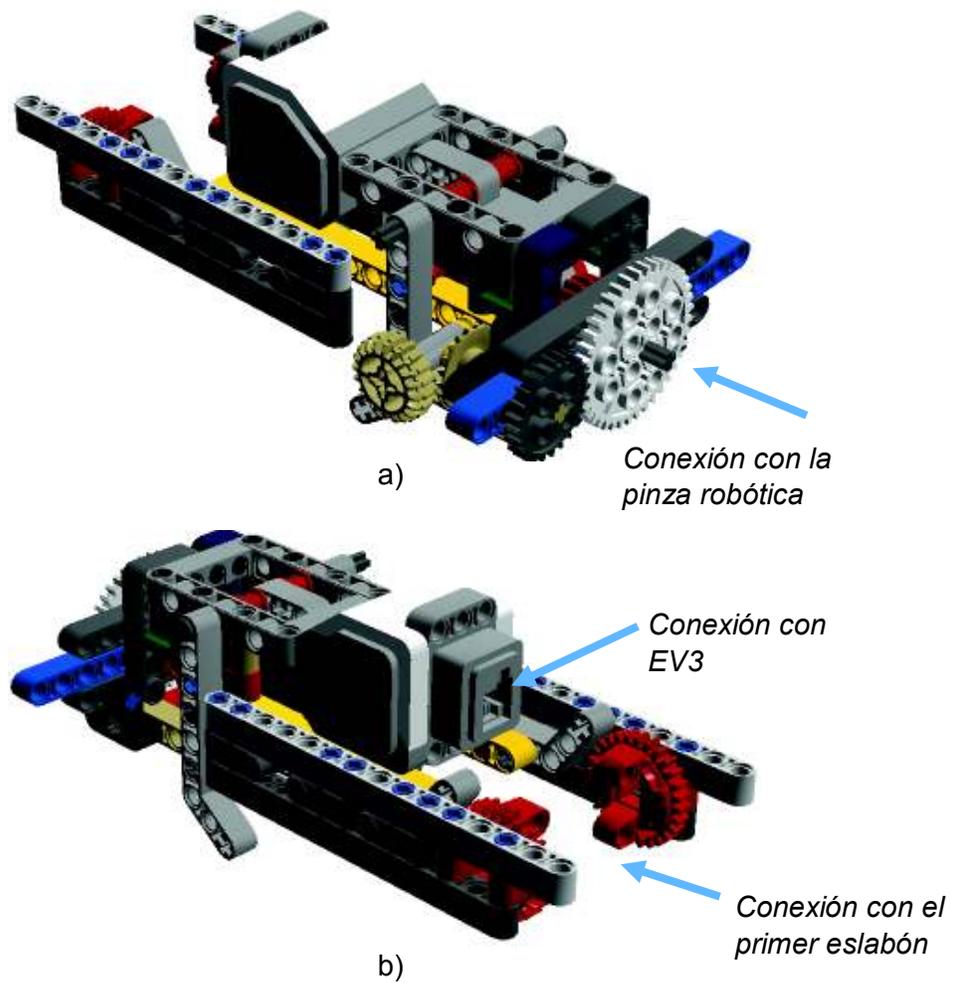


Figura 2.33 Segundo eslabón del brazo robótico. a) vista frontal, b) vista trasera

Construcción de la pinza robótica

La pinza del brazo robótico implementado es una modificación de un diseño público [57]. Esta pinza utiliza el único motor mediano, m_4 , conectado al Puerto D del ladrillo inteligente.



Figura 2.34 Modelo de pinza robótica [57]

La modificación del diseño público de [57] se realizó porque la pinza era sumamente larga, lo que inclinada al robot en la articulación Q_2 . La modificación consiste en cambiar el eje de giro del motor mediano, m_4 , tal que su longitud se reduzca casi en un tercio. En la Figura 2.35 se presenta la pinza del brazo robótico.

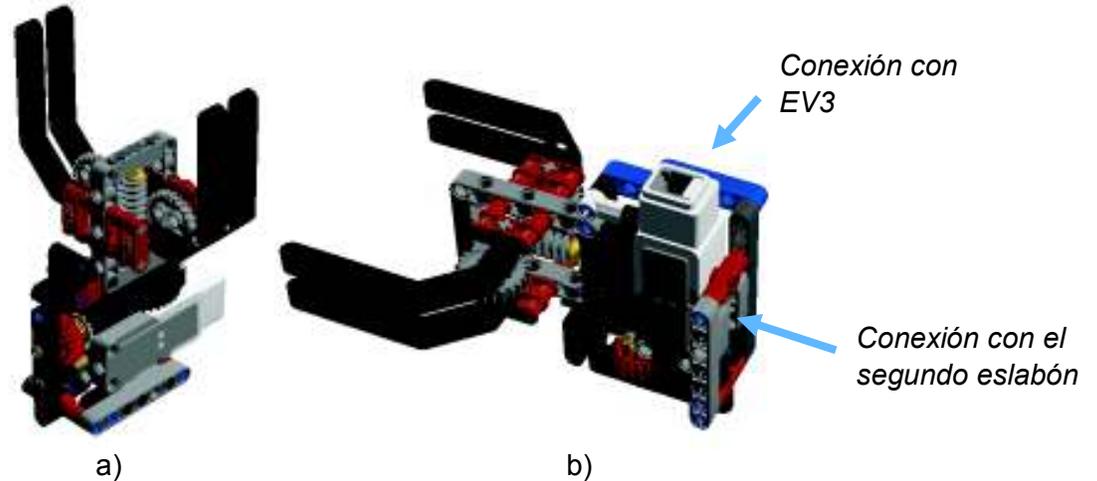


Figura 2.35 Pinza del brazo robótico a) vista inferior, b) vista lateral

Las cuatro partes antes descritas se juntan para formar el brazo robótico. La base donde se asienta el robot va conectada al primer eslabón, del primer eslabón se conecta el segundo en un ángulo de 90° , (ver sección 0). Finalmente se coloca la pinza al extremo final del segundo eslabón. En la Figura 2.36, se presenta el brazo robótico implementado.

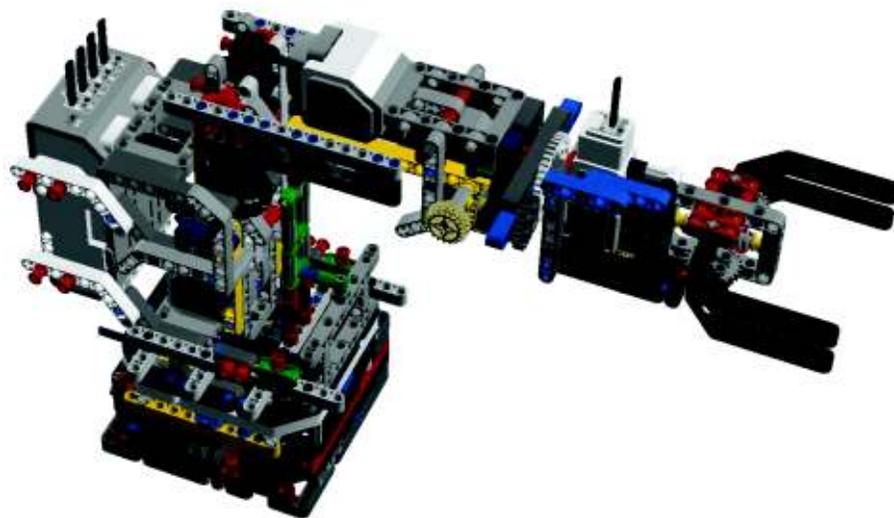


Figura 2.36 Brazo robótico implementado

Se establece límites para los ángulos de giro sobre cada grado de libertad. Estas restricciones se establecen a partir del modelo de brazo robótico implementado. En la Tabla 2.8 se presenta los ángulos límites para cada grado de libertad del brazo robótico.

Tabla 2.8 Restricciones para cada grado de libertad

Grado de libertad	Límite inferior [deg]	Límite superior [deg]
<i>Primero</i>	-75	75
<i>segundo</i>	-30	60
<i>tercero</i>	-135	135

Para el correcto comando del brazo robótico, cada uno de los motores M_1 , M_2 , M_3 y m_4 se debe conectar al respectivo puerto del ladrillo inteligente EV3 como se muestra en la Figura 2.37.

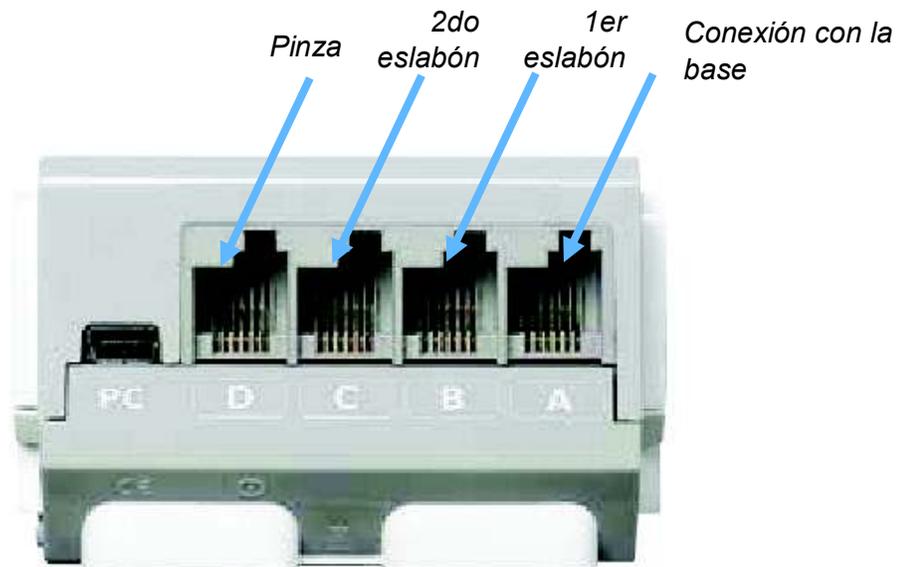


Figura 2.37 Conexión de los motores con el ladrillo inteligente EV3

Codificación de los gestos para acciones del robot

Los gestos reconocidos por el sistema de clasificación son codificados en comandos que se envían vía Bluetooth al ladrillo EV3. Se asigna a cada gesto un comando específico del robot. En la Tabla 2.9 se presenta los comandos correspondientes a cada gesto.

Tabla 2.9 Comandos correspondientes a cada gesto

Gesto	Comando
<i>Fist</i>	<i>cerrar pinza</i>
<i>Open</i>	<i>abrir pinza</i>
<i>Wave Out</i>	<i>Detener ejecución</i>
<i>Pinch</i>	<i>volver a posición inicial</i>
<i>Wave In</i>	<i>iniciar y pausar ejecución</i>

Se escogió el gesto *Open* y el gesto *Fist* para el control de la pinza porque estos gestos son opuestos el uno al otro, siendo los gestos que más se asemejan a la apertura y cierre de la pinza robótica. El gesto *Wave In* inicia y pausa la ejecución del mando del brazo robótico; el gesto *Wave Out* se usa únicamente para detener completamente la ejecución del sistema. Mientras que, el gesto *Pinch*, sirve para mover el brazo robótico a su posición original.

Señales de referencia para la posición angular de cada grado de libertad

Las señales de referencia para cada articulación del brazo robótico son los ángulos de Tait-Bryan: alabeo, cabeceo y guiñada, conocidos también en inglés como *yaw*, *pitch*, *roll*, respectivamente. Estos ángulos son calculados a partir de la señal de orientación del Myo Armband (ver sección 0).

Los ángulos de Tait-Bryan tienen la desventaja del Bloqueo del cardán. El bloqueo del cardán es una configuración en la que el sistema pierde un grado de libertad cuando el ángulo de cabeceo es 90° [58]. A pesar de esta dificultad, se escoge utilizar los ángulos de Tait-Bryan sobre cuaternios por su facilidad de interpretación. Se escoge limitar el espacio de trabajo para no incurrir en el Bloqueo del cardán.

Es importante mencionar aquí que el Myo Armband retorna la orientación con respecto a los ejes de la Tierra. Es decir, utilizando solo una unidad de medición inercial IMU (Myo Armband), solo se puede distinguir la orientación del antebrazo con respecto a un eje coordenado fijo. Para el caso del presente estudio técnico los ángulos de alabeo, cabeceo y guiñada son definidos para el brazo estirado, como se muestra en la Figura 2.38.

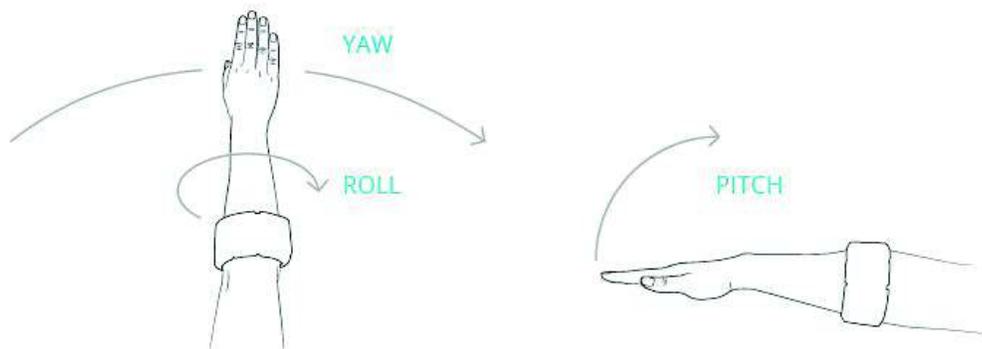


Figura 2.38 Definición de los ángulos guiñada (*yaw*), cabeceo (*pitch*) y alabeo (*roll*) para el Myo Armband [34]

Conversión de Matrices de rotación a ángulos de Tait-Bryan [58]

El presente estudio técnico utiliza el Myo Armband para la adquisición de las señales EMG y la señal de orientación. La señal de orientación que retorna el Myo Armband se expresa como una matriz de rotación. En el presente apartado se explica la conversión de la matriz de rotación a los ángulos de Tait-Bryan.

Una matriz de rotación es una matriz ortonormal que rota a un vector definido con respecto a un marco de referencia $\{B\}$ hacia un vector con respecto a un marco de referencia $\{A\}$.

$$\begin{bmatrix} A_x \\ A_y \\ A_z \end{bmatrix} = {}^A R_B \begin{bmatrix} B_x \\ B_y \\ B_z \end{bmatrix}$$

Ecuación 2.28 Definición de la matriz de rotación

La matriz de rotación, ${}^A R_B$, o simplemente, R , es una matriz donde sus filas y columnas tienen magnitud unitaria. La matriz de rotación contiene 3 valores independientes. Se define una matriz de rotación para cada eje coordenado.

$$R_x(\theta) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{pmatrix} \quad R_y(\theta) = \begin{pmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{pmatrix} \quad R_z(\theta) = \begin{pmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Ecuación 2.29 Matrices de rotación para cada eje coordenado R_x, R_y, R_z

El teorema de la rotación de Euler establece que cualquier rotación puede ser expresada por 3 rotaciones alrededor de los ejes coordenados. La convención de giros xyz, guiñada, cabeceo, alabeo, (*yaw, pitch, roll*) es conocida como ángulos de Cardan, ángulos de Tait-Bryan o ángulos náuticos.

$$R = R_x(\theta_r)R_y(\theta_p)R_z(\theta_y)$$

Ecuación 2.30 Ángulos de Tait-Bryan

Donde: θ_r es el ángulo de alabeo (*roll*)

θ_p es el ángulo de cabeceo (*pitch*)

θ_y es el ángulo de guiñada (*yaw*)

De las tres matrices de rotación para cada eje coordenado se obtiene la matriz de rotación total.

$$R = \begin{pmatrix} \cos \theta_y \cos \theta_p & -\cos \theta_p \sin \theta_y & \sin \theta_p \\ \cos \theta_r \sin \theta_y + \cos \theta_y \sin \theta_p \sin \theta_r & \cos \theta_y \cos \theta_r - \sin \theta_y \sin \theta_p \sin \theta_r & -\cos \theta_p \sin \theta_r \\ \sin \theta_y \sin \theta_r - \cos \theta_y \cos \theta_r \sin \theta_p & \cos \theta_r \sin \theta_r + \cos \theta_r \sin \theta_y \sin \theta_p & \cos \theta_p \cos \theta_r \end{pmatrix}$$

Ecuación 2.31 Matriz de rotación en función de los ángulos de Tait-Bryan

De la Ecuación 2.31 se encuentra los valores de los ángulos de guiñada θ_y , cabeceo θ_p , y alabeo θ_r .

$$\theta_y = \text{atan2}(R_{(1,2)}, R_{(1,1)}), \quad \theta_p = \text{atan2}\left(R_{(1,3)}, \sqrt{R_{(2,3)}^2 + R_{(3,3)}^2}\right), \quad \theta_r = \text{atan2}(-R_{(2,3)}, R_{(3,3)})$$

Ecuación 2.32 Ángulos de Tait-Bryan: guiñada θ_y , cabeceo θ_p , y alabeo θ_r

Donde: $\text{atan2}(y, x)$ es la función arco tangente para dos argumentos de entrada. Esta función es el arco tangente en 4 cuadrantes.

De la matriz de rotación que retorna el Myo Armband se calcula los ángulos de Tait-Bryan (Ecuación 2.32). Estos ángulos son las señales de referencia para cada controlador de posición angular del brazo robótico implementado.

Ángulos de Tait-Bryan para el movimiento del Myo Armband

Se realizaron mediciones del movimiento del brazo utilizando el Myo Armband. Los movimientos realizados fueron alrededor de cada uno de los ángulos Tait-Bryan. En la Figura 2.39, Figura 2.40 y Figura 2.41 se presenta tres repeticiones de movimiento alrededor del ángulo de guiñada, cabeceo y alabeo, respectivamente.

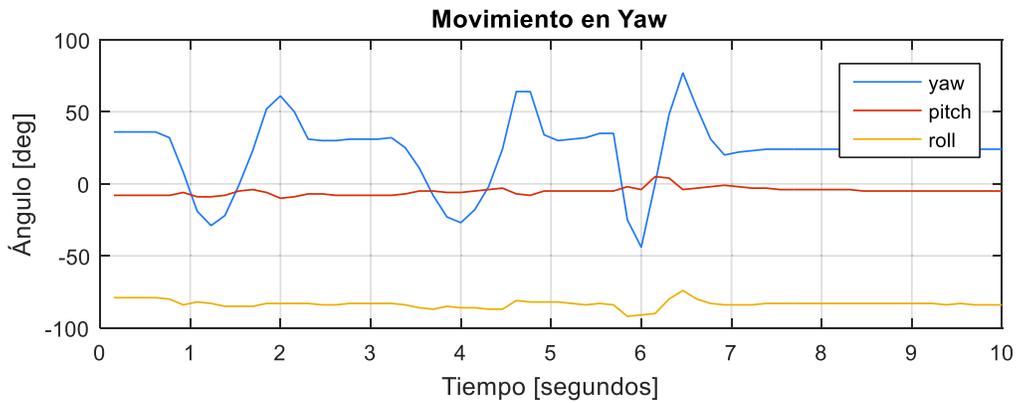


Figura 2.39 Movimiento del brazo alrededor del ángulo de guiñada

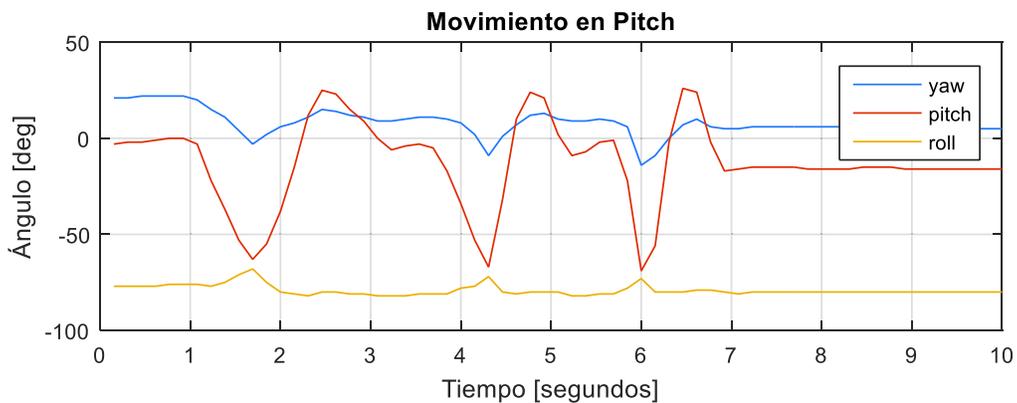


Figura 2.40 Movimiento del brazo alrededor del ángulo de cabeceo

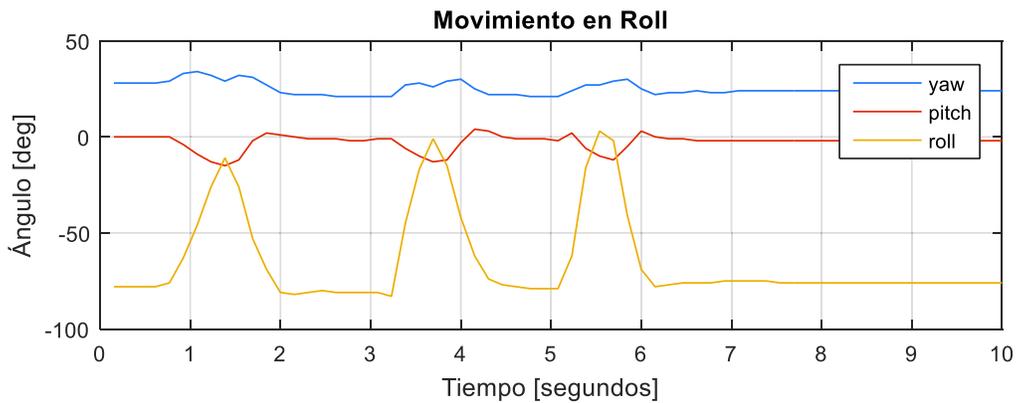


Figura 2.41 Movimiento del brazo alrededor del ángulo de alabeo

En estas mediciones se realizaron tres repeticiones a diferentes velocidades. Estas se perciben claramente en cada ángulo de movimiento. Sin embargo, existen oscilaciones de los demás ángulos, especialmente en el caso del ángulo de cabeceo, donde el ángulo de guiñada oscila alrededor de 15 grados Figura 2.40. En las siguientes figuras se muestran las mediciones realizadas con el Myo Armband cuando no está ubicado en el brazo (sobre una superficie plana).



Figura 2.42 Movimiento del Myo Armband alrededor del ángulo de guiñada



Figura 2.43 Movimiento del Myo Armband alrededor del ángulo de cabeceo

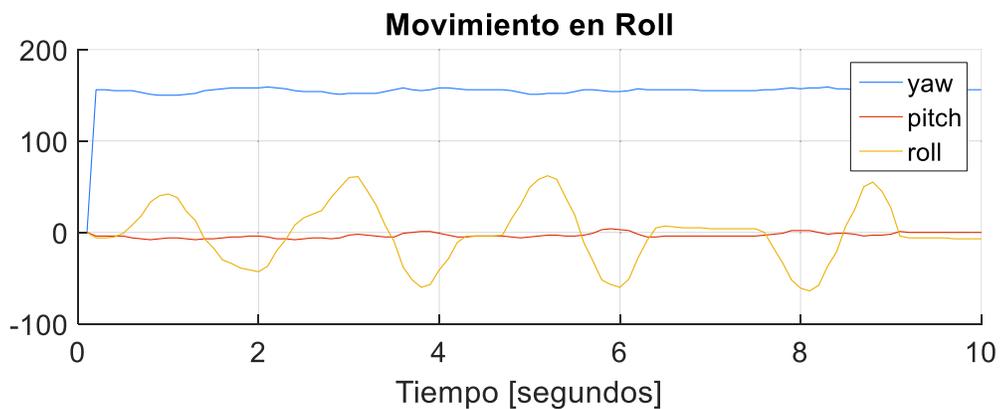


Figura 2.44 Movimiento del Myo Armband alrededor del ángulo de alabeo

La comparación de ambas mediciones del movimiento del Myo Armband demuestra que al utilizarse el Myo Armband en el brazo del usuario aparecen ligeras oscilaciones en todos los ángulos. Estas oscilaciones pueden ser generadas debido a la naturaleza de movimiento propia del brazo.

De estas gráficas también se observa que tienen un *offset*. Más adelante se explicará que, para las señales de referencia, se tomará como posición inicial a la primera medición de movimiento.

Controladores PID en las articulaciones del brazo robótico de 3 GDL

Cada grado de libertad del brazo robótico posee un control PID de posición angular. Como se describió en la sección 0, las señales de referencia son los ángulos de Tait-Bryan: guiñada, cabeceo y alabeo (*yaw*, *pitch* y *roll*, respectivamente). La forma análoga ideal del controlador PID se muestra en la Ecuación 2.33.

$$G_c(s) = \frac{U(s)}{E(s)} = K_p \left(1 + \frac{1}{t_i s} + t_d s \right)$$

Ecuación 2.33 Controlador PID [59]

Donde: K_p es la ganancia proporcional

t_i es el tiempo integral

t_d es el tiempo derivativo

$U(s)$ es la transformada de Laplace de la señal de control $u(t)$

$E(s)$ es la transformada de Laplace para la señal de error $e(t) = r(t) - y(t)$, con $r(t)$ como señal de referencia y $y(t)$ como señal de salida

A partir de la Ecuación 2.33, se encuentra el controlador PID digital. Para ello, se aproxima la componente integral del controlador a través de la sumatoria trapezoidal; y la componente derivativa se aproxima como diferencia hacia atrás [60].

$$G_c(z) = \frac{U(z)}{E(z)} = K'_p + \frac{K_I}{1 - z^{-1}} + K_D(1 - z^{-1})$$

Ecuación 2.34 Controlador PID digital de la forma posicional [60]

Donde: $K'_p = K_p - \frac{K_p T}{2t_i}$ es la ganancia proporcional digital

$K_I = \frac{K_p T}{t_i}$ es la ganancia integral

$$K_D = \frac{K_p T_d}{T} \text{ es la ganancia derivativa}$$

T es el periodo de muestreo

Se prefiere la forma posicional del controlador PID en lugar de la forma de velocidad porque la forma de velocidad necesita de la segunda derivada del error [61]. Del controlador PID digital de la Ecuación 2.34, se encuentra la respectiva ley de control.

$$u[nT] = u[(n-1)T] + e[nT] * (K'_p + K_I + K_D) - e[(n-1)T] * (K'_p + 2K_D) + e[(n-2)T]$$

Ecuación 2.35 Ley de Control del controlador PID de la forma posicional

Calibración de los controladores PID

Las constantes del controlador PID fueron calibradas basándose en el método descrito en [62]. Este procedimiento es un método iterativo que varía una por una las constantes del controlador, ver Figura 2.45. Se registra la respuesta del sistema para cada iteración. La respuesta del sistema es la posición angular de cada grado de libertad. Se realizó el procedimiento hasta conseguirse una respuesta satisfactoria. Esta evaluación se realizó visualmente.

Se utilizó una señal patrón como señal de referencia para cada grado de libertad al ejecutar el proceso de calibración. La señal patrón fue adquirida utilizando el Myo Armband; esta señal patrón es una secuencia de movimientos que incluye un movimiento:

- izquierda—derecha en *yaw*,
- arriba—abajo en *pitch*,
- horario—antihorario en *roll* y
- un movimiento combinado de los tres (círculo con el brazo)

De esta secuencia de movimientos adquirida se extrae las 3 componentes (*yaw*, *pitch* y *roll*) a ser utilizadas como señales de referencia, ver Figura 2.46.

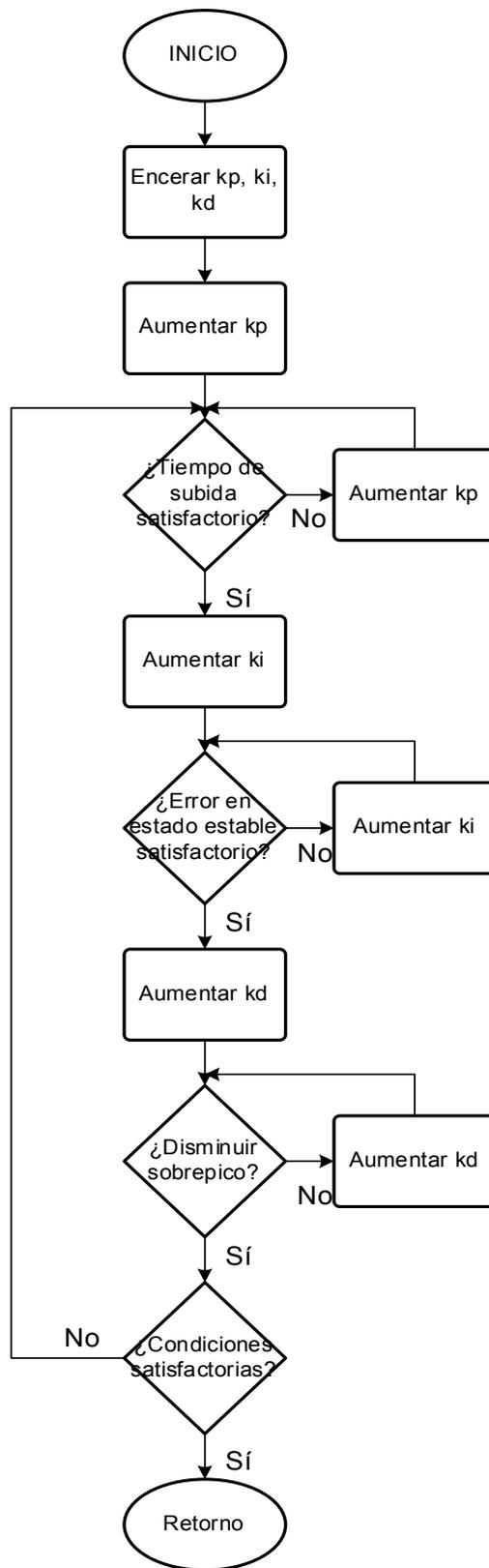


Figura 2.45 Diagrama de Flujo calibración de los controladores PID [62]

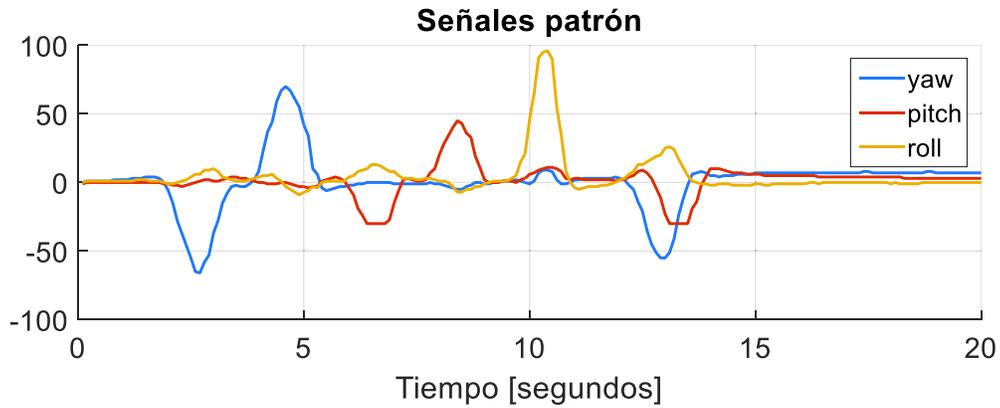


Figura 2.46 Señales patrón para la calibración de los PID

A continuación se presenta el procedimiento realizado para la calibración de los controladores. Para facilidad de visualización se reportan un número reducido de iteraciones.

Calibración del primer grado de libertad (movimiento en yaw)

Se inicializó el proceso de calibración del primer grado de libertad variando el valor de la constante proporcional. En la Figura 2.47 se muestra la respuesta para diferentes valores de K'_p .

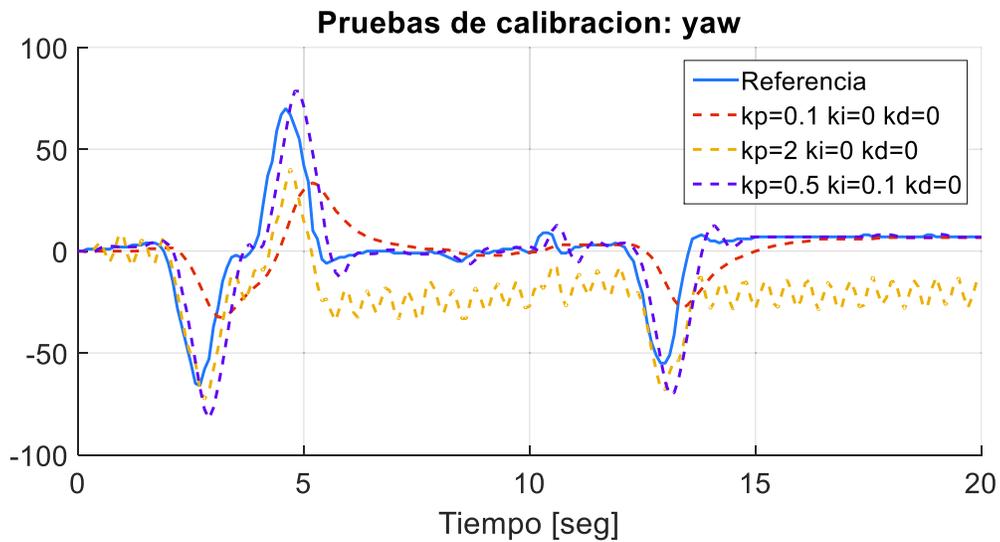


Figura 2.47 Calibración del primer grado de libertad (1)

Para valores de la constante proporcional mayores a 0.5 se vuelve oscilatorio el sistema. En la Figura 2.48 se procede a modificar la constante integradora.

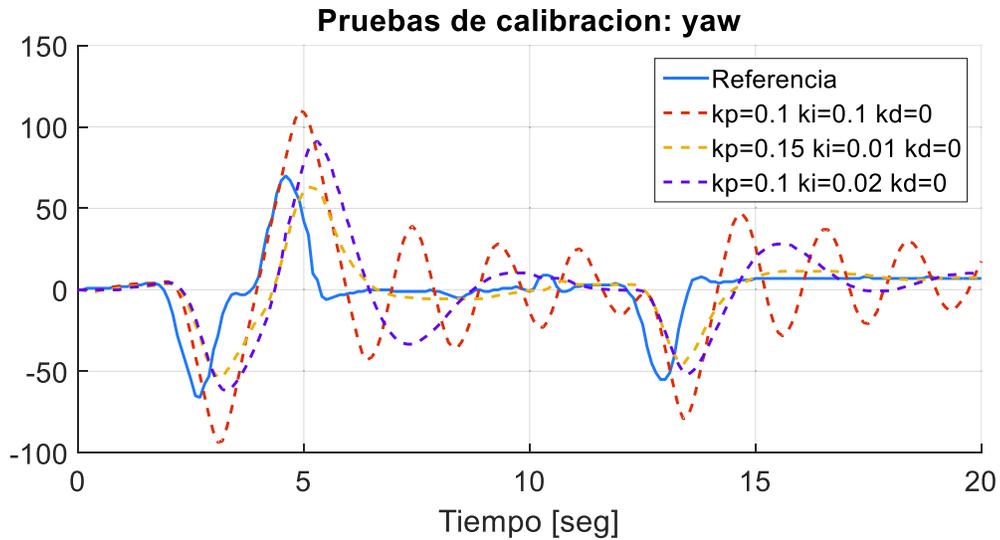


Figura 2.48 Calibración del primer grado de libertad (2)

Se observa en la Figura 2.48 que cuando la constante integral es 0.1 el sistema oscila notablemente. Se procede a realizar pruebas para valores de 0.05 y 0.015.

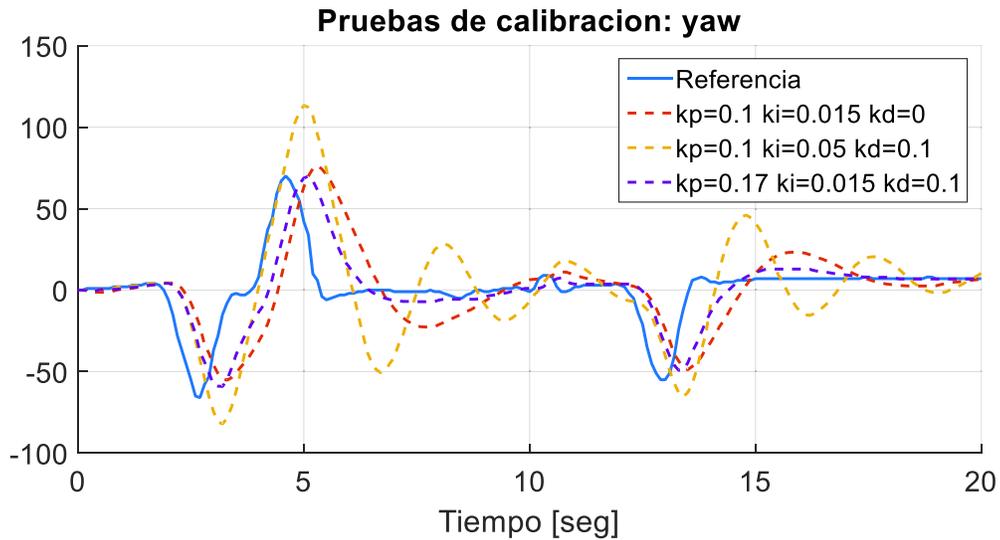


Figura 2.49 Calibración del primer grado de libertad (3)

De la figura se observa que la constante integral 0.05 produce una respuesta oscilatoria. Se establece K_I en 0.015. En la Figura 2.50, se procede a probar diferentes valores de la constante derivativa. En la Tabla 2.10 se resume las constantes escogidas para el primer grado de libertad.

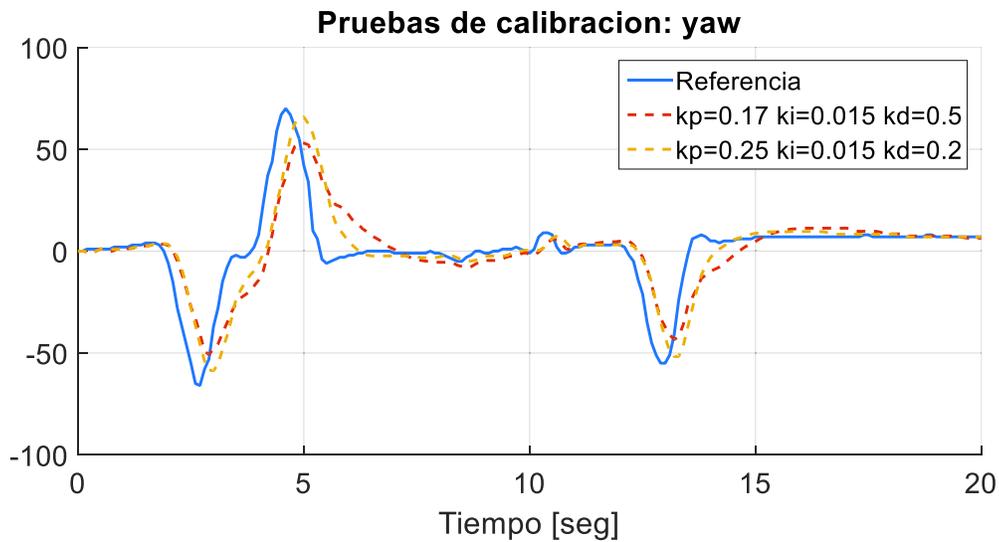


Figura 2.50 Calibración del primer grado de libertad (4)

Tabla 2.10 Constantes del controlador PID para el primer grado de libertad

Parámetro	Valor
K'_p	0.25
K_I	0.015
K_D	0.2

Calibración del segundo grado de libertad (movimiento en pitch)

Para calibrar el controlador PID del segundo grado de libertad se realizó un procedimiento similar al anterior. En este caso, el procedimiento fue más corto, el número de pruebas requeridas fue menor para llegar a una respuesta satisfactoria.

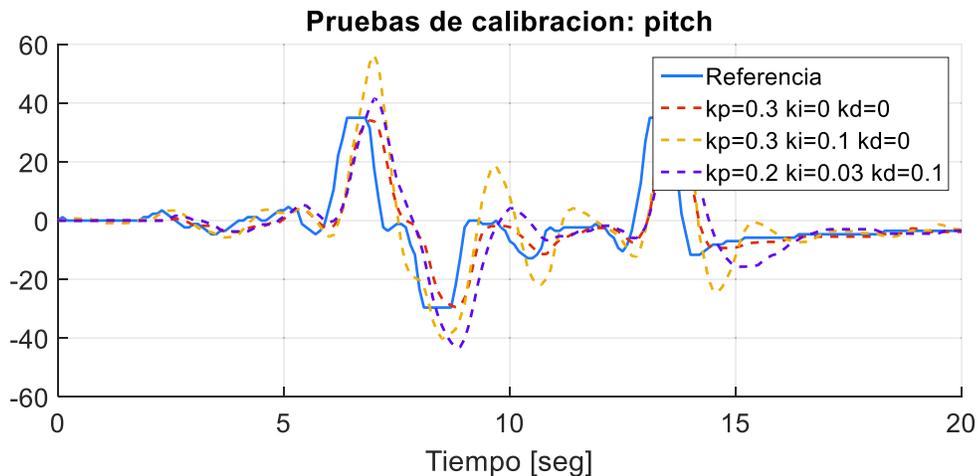


Figura 2.51 Calibración del segundo grado de libertad (1)

En la Figura 2.51 se observa que al tener la ganancia proporcional igual a 0.3 y la ganancia integral a 0.1 se obtiene una respuesta oscilatoria. En la Figura 2.52 se continúa el proceso de calibración.

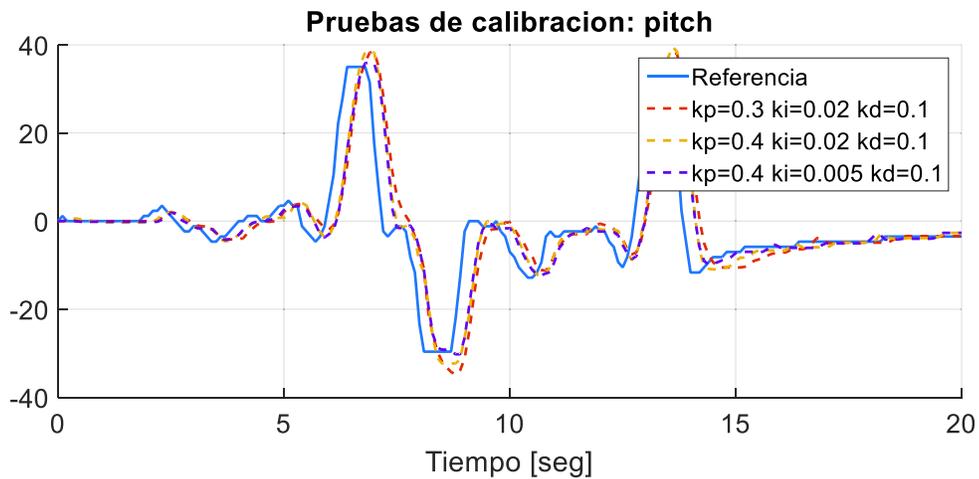


Figura 2.52 Calibración del segundo grado de libertad (2)

Se observa que el procedimiento de calibración obtuvo respuestas similares. De estas respuestas se establece los parámetros para el controlador. Los parámetros establecidos para el controlador PID del segundo grado de libertad se resumen en la Tabla 2.11.

Tabla 2.11 Constantes del controlador PID del segundo grado de libertad

Parámetro	Valor
K'_p	0.4
K_I	0.02
K_D	0.1

Calibración del tercer grado de libertad (movimiento en roll)

De manera similar se calibró el controlador PID del tercer grado de libertad. Se inicializó el controlador como tipo P. De ahí se varió la ganancia proporcional hasta obtener un tiempo de subida adecuado.

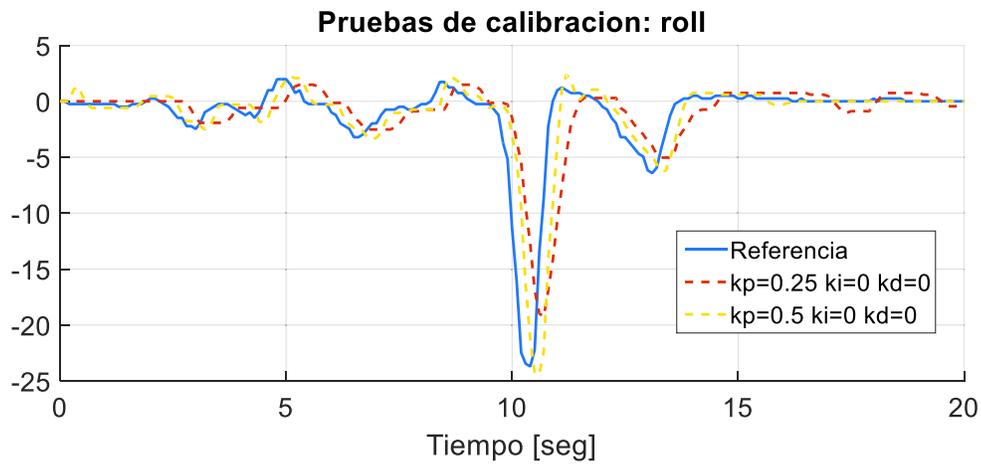


Figura 2.53 Calibración del tercer grado de libertad (1)

Para los parámetros de K'_p menores a 0.25 el controlador responde lentamente; en cambio, para valores mayores a 0.5, el sistema tiene un sobrepico indeseado. Para las siguientes iteraciones se modifica el parámetro de la ganancia integradora K_I .

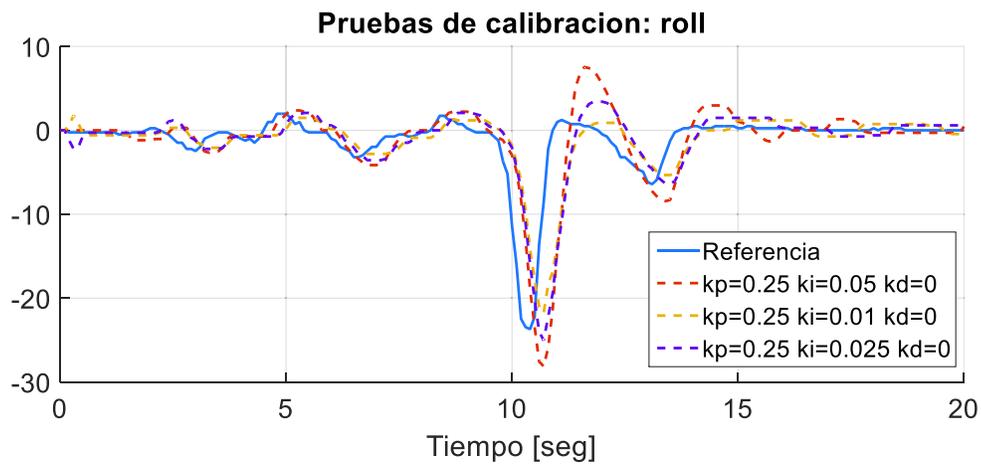


Figura 2.54 Calibración del tercer grado de libertad (2)

Al variar la ganancia integradora se obtiene una respuesta oscilatoria. Se escoge K_I igual a 0.025.

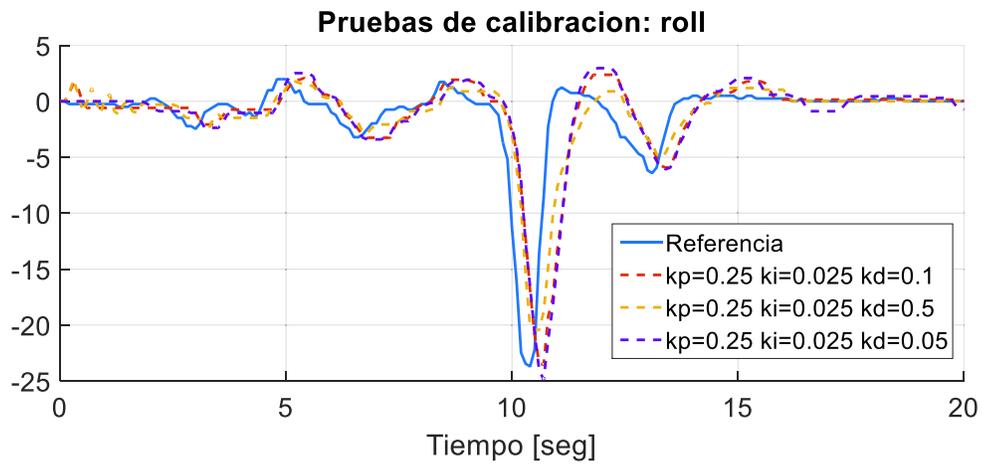


Figura 2.55 Calibración del tercer grado de libertad (3)

Para disminuir el sobrepico, se varía el valor de la ganancia derivativa. Los parámetros establecidos para el controlador PID del tercer grado de libertad se resumen en Tabla 2.12.

Tabla 2.12 Constantes del controlador PID del tercer grado de libertad

Parámetro	Valor
K'_p	0.25
K_I	0.025
K_D	0.1

2.5 Implementación de la interfaz de usuario en Matlab

Se implementó una interfaz gráfica en Matlab R2015a para este proyecto, Figura 2.56. Se utiliza la función para trabajar con lazos en paralelo de Matlab *parpool*, además, se incluye validación de datos para todos los campos. En el ANEXO III. MANUAL DE USUARIO se encuentra el manual de usuario detallado de la interfaz gráfica. En el ANEXO IV. DIAGRAMAS DE FLUJO se incluye los diagramas de flujo del programa implementado.

Esta interfaz gráfica permite al usuario:

- Conectarse y desconectarse con el Myo Armband y con el ladrillo inteligente EV3,
- Visualizar la orientación y las señales EMG del Myo Armband,
- Realizar la rutina de entrenamiento,
- Visualizar los gestos adquiridos de la rutina de entrenamiento,
- Visualizar el resultado del sistema de clasificación de gestos,
- Iniciar, pausar y detener el mando a distancia del brazo robótico.



Figura 2.56 Interfaz gráfica de usuario

Para ejecutar el sistema se debe conectar el Myo Armband; al conectar el Myo Armband, se muestran en pantalla la orientación y las señales EMG. De ahí es necesario que el usuario realice la rutina de entrenamiento. En esta interfaz, Figura 2.57, se debe realizar la adquisición de cada una de las repeticiones de los gestos. Como se estableció en el apartado 192.2, la rutina de entrenamiento consiste en 5 repeticiones durante 2 segundos. Se

muestra al usuario una barra del progreso del tiempo y el número de repeticiones realizadas.



Figura 2.57 Interfaz gráfica para la rutina de entrenamiento

Después de realizar la rutina de entrenamiento se puede visualizar las señales EMG de los gestos grabados, Figura 2.58. En esta interfaz se muestra las señales EMG puras y las señales EMG al aplicarse el filtro. Se puede seleccionar qué gesto y a qué repeticiones mostrar en el gráfico.



Figura 2.58 Interfaz gráfica para visualización de los gestos realizados en el entrenamiento

Para iniciar el comando del brazo robótico se debe conectar con el ladrillo inteligente EV3. Para ello se debe ejecutar el programa “fullPID” en el EV3 y después presionar en “CONECTAR LEGO”. Al tener listas las conexiones se puede habilitar el mando del brazo robótico, botón “INICIAR”. Al realizar esto, el sistema de clasificación empieza a ejecutarse, y devuelve una imagen del gesto predicho (en el caso de que el gesto predicho sea **reposo** no se muestra imagen alguna). Las gráficas de la orientación y de las señales EMG se deshabilitan para liberar recursos para el reconocimiento.

3. RESULTADOS Y DISCUSIÓN

De acuerdo con los objetivos del presente proyecto, se realizan pruebas para validar el funcionamiento del sistema de clasificación de gestos y del mando a distancia del brazo robótico de 3 GDL. El sistema de clasificación de gestos debe cumplir con dos parámetros. Estos parámetros son: la tasa de exactitud de clasificación; y el tiempo de respuesta del sistema. Se planteó como alcance, que la tasa de exactitud de clasificación debe ser de al menos 85%; mientras que, el tiempo de respuesta del sistema debe ser menor a 300 milisegundos. En el caso del brazo robótico de 3 GDL, se estableció que debe tener un controlador PID de la posición angular sobre cada grado de libertad. El controlador PID debe cumplir con un error menor a 5°.

3.1 Tasa de clasificación y Tiempo de respuesta del sistema de clasificación de gestos

Para evaluar la exactitud de clasificación, y el tiempo de respuesta del sistema, se recolectaron las señales EMG de diez usuarios. A cada usuario se solicitó que realice la rutina de entrenamiento, y que realice 30 repeticiones adicionales de cada gesto (en total, 35 repeticiones: 5 para entrenamiento y 30 para evaluación). En el caso de las repeticiones para evaluación, el usuario podía realizar el gesto dentro de una ventana de tiempo de 5 segundos. Esta ventana de tiempo es mayor a la utilizada para el entrenamiento (de 2 segundos). Se estableció una ventana de mayor duración con el propósito de evaluar el sistema en tiempo real (el sistema debe estar en capacidad de descartar fragmentos de señal que no correspondan al gesto). En la Tabla 3.1 se resume las condiciones de prueba para evaluar el sistema de clasificación de gestos.

Tabla 3.1 Condiciones de prueba para el sistema de clasificación de gestos

Parámetro	Valor
<i>Usuarios</i>	10
<i>Repeticiones</i>	30
<i>Tiempo [segundos]</i>	5

El sistema de clasificación de gestos se evaluó sobre las 30 repeticiones de las señales EMG de la rutina de evaluación. Debido a que la clasificación es en tiempo real, el sistema retorna resultados consecutivos cada 200 milisegundos. Es decir que, para cualquier repetición de 5 segundos, se obtienen 25 resultados consecutivos. Fue necesario establecer un

protocolo de evaluación que determine un único resultado entre todos los resultados obtenidos de la ventana de 5 segundos de cada repetición. El protocolo se estableció considerando la etapa de posfiltrado de la señal (ver sección 0), y se definió de la siguiente manera:

- Cuando todas las clasificaciones son el gesto “reposo”, el gesto resultante es “reposo”.
- Cuando entre las clasificaciones exista un solo gesto diferente al gesto “reposo”, este se considera el gesto resultante.
- Cuando entre las clasificaciones exista más de un gesto diferente al gesto “reposo”, el gesto resultante es el más común de los gestos resultantes.

Con el protocolo de evaluación establecido, se ejecutó el sistema de clasificación de gestos sobre las señales de los diez usuarios que participaron en la rutina de pruebas. En la Figura 3.1 se muestra la matriz de confusión con los resultados de todos los usuarios. En la Figura 3.2 se presenta el porcentaje de gestos clasificados correctamente por usuario.

Matriz de Confusión

		noGesto	Wave In	Wave Out	Fist	Open	Pinch	
Predicción del sistema	no Gesto	0 0.0%	9 0.6%	36 2.4%	13 0.9%	32 2.1%	35 2.3%	0.0% 100%
	Wave In	0 0.0%	278 18.5%	0 0.0%	5 0.3%	1 0.1%	4 0.3%	96.5% 3.5%
	Wave Out	0 0.0%	4 0.3%	255 17.0%	1 0.1%	0 0.0%	0 0.0%	98.1% 1.9%
	Fist	0 0.0%	1 0.1%	0 0.0%	252 16.8%	1 0.1%	0 0.0%	99.2% 0.8%
	Open	0 0.0%	3 0.2%	5 0.3%	17 1.1%	250 16.7%	6 0.4%	89.0% 11.0%
	Pinch	0 0.0%	5 0.3%	4 0.3%	12 0.8%	16 1.1%	255 17.0%	87.3% 12.7%
			92.7% 7.3%	85.0% 15.0%	84.0% 16.0%	83.3% 16.7%	85.0% 15.0%	86.0% 14.0%
		noGesto	Wave In	Wave Out	Fist	Open	Pinch	
		Gesto Realizado						

Figura 3.1 Matriz de confusión con los resultados de la clasificación

Las columnas de la matriz de confusión representan el gesto realizado, mientras que las filas, representan la respuesta del sistema. Cada gesto fue realizado 300 veces en total (30 repeticiones hechas por diez usuarios), por lo que cada columna debe sumar 300 (en total serían 1500 gestos; 300 repeticiones cada gesto, 5 gestos realizados). En el caso de las filas, estas no deben sumar necesariamente 300, pues un gesto pudo haber sido detectado por el sistema más veces de las que fue realizado.

Las últimas fila y columna representan el porcentaje de acierto (texto en verde), y el porcentaje de error (texto en rojo) de cada gesto. Estos dos resultados corresponden a porcentajes de acierto, pero de una manera diferente. Los resultados de la última fila (fondo gris) son el porcentaje de clasificados correctamente entre todos los gestos realizados correspondientes a esa categoría. En cambio, los resultados de la última columna son el porcentaje de clasificados correctamente entre el conjunto de gestos que el sistema reconoció pertenecientes a esa categoría. Por ejemplo, en los resultados de la primera fila, correspondientes al gesto “reposo” o “no Gesto”, se muestra los valores 0% y 100%, esto quiere decir que cero por ciento de acierto de las predicciones del sistema que devolvieron “no Gesto” fueron correctas. Este resultado es obvio, pues en ninguna repetición se pretendió hacer ese gesto, por lo que todas las detecciones fueron errores. Si analizamos el caso del gesto *Fist*, tiene un 99.2% por ciento en detección, pues 252 clasificaciones fueron correctas de las de las 254 veces que fue detectado. En cambio, el mismo gesto *Fist*, tiene un 84% de clasificación, pues fue reconocido correctamente 252 veces, de las 300 veces que fue realizado. Es decir, en la última fila se presenta el resultado de cuán bien clasificó el sistema a un gesto entre los realizados; mientras que, en la última columna, se presenta el resultado de cuán bien clasificó a un gesto entre las veces que ese gesto fue detectado.

En las demás casillas se presenta el número de clasificaciones en cada caso, con su respectivo porcentaje. Por ejemplo, En la segunda casilla de la primera fila se presenta los valores 9 y 0.6%, pues 9 gestos *Wave In* fueron clasificados como “no Gesto”, eso corresponde al 0.6 % de todo el conjunto de gestos (9 de 1500). La diagonal (fondo verde), idealmente debería ser 300 y 20%, pues cada uno de los cinco gestos (20% cada uno) se repitió 300 veces. Los valores de la primera columna son todo cero, pues nunca se realizó el gesto “reposo” entre las repeticiones.

La matriz de confusión muestra que el gesto mejor clasificado es el gesto *Wave In*, (92.7%) por un gran margen con respecto a los demás gestos (que están alrededor del 84%). Los gestos *Wave In*, *wave Out* y *Fist*, tienen altos porcentajes de detección (96.5%,98.1%,99.2% respectivamente). Vemos que el gran déficit del sistema está en las no detecciones; en muchas repeticiones el sistema devolvió como respuesta el gesto “reposo”. En el caso de los gestos *Wave Out*, *Open* y *Pinch*, más de 30 veces el sistema los confundió con “no Gesto” (36,32 y 35 veces, respectivamente, que suman alrededor de 100 repeticiones). El sistema también presenta confusiones entre los gestos: 17 veces el sistema confundió el gesto *Fist* con el gesto *Open*; 12 veces con el gesto *Pinch*. También fueron 16 veces que el gesto *Open* se confundió con el gesto *Pinch*. Aunque estos resultados, (1.1%, 0.8%, 1.1% respectivamente) son ligeramente altos, son mucho menores a

las no detecciones (2.4%, 2.1%, 2.3%). El sistema presenta una precisión de clasificación promedio del 86%.

Al analizar los resultados promedios de cada usuario, Figura 3.2, vemos que existe una gran variación entre las personas voluntarias de las pruebas. Con valores sumamente altos, 97.3% y 94.7% para los usuarios *andres* y *juan*; pero también bajos, como en el caso de los usuarios *homero*, y *dianita*, 74.7% y 78.7%, respectivamente. Cabe también mencionar que los usuarios que participaron en varias pruebas del sistema son quienes obtienen mejores resultados; mientras que, aquellos que lo probaron por primera vez, obtuvieron los peores. Esto demuestra que la precisión de clasificación aumenta significativamente con la práctica.

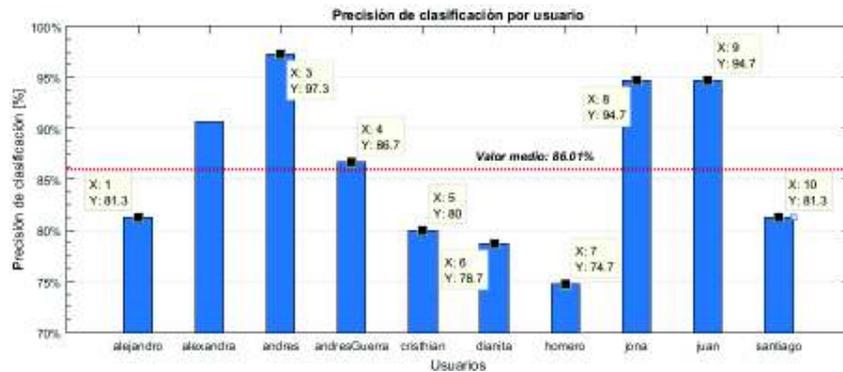


Figura 3.2 Precisión de clasificación por usuario

Las matrices de confusión por usuario, junto con mayores detalles de la rutina de pruebas se incluyen en el ANEXO V. RUTINA DE PRUEBAS.

De estas mismas pruebas se midieron tiempos de respuesta del sistema. Sobre este aspecto es importante mencionar que las características del computador utilizado afectan la velocidad del sistema. En el presente estudio técnico, también parte del Proyecto Junior “Clasificación de señales electromiográficas del brazo humano usando técnicas de reconocimiento de patrones y machine learning”, se utilizó un computador de escritorio con un procesador Intel Core™ i7-3770S 4 núcleos, 64 bits, de 4 Gb de memoria RAM. Un histograma con los tiempos de procesamiento de todas las clasificaciones se muestra en la Figura 3.3.



Figura 3.3 Histograma de tiempos de procesamiento

Este histograma considera las clasificaciones consecutivas en tiempo real de todas las repeticiones. Como el sistema en tiempo real devuelve resultados en menos de 300 milisegundos, y cada repetición de la sesión de pruebas dura 5 segundos, son aproximadamente cuatro clasificaciones por segundo, 20 clasificaciones por repetición. Por lo que en las 15000 repeticiones se obtienen alrededor de 30000 clasificaciones. Se ve que las clasificaciones no duran más de 300 milisegundos, pero tampoco son menores a 175 milisegundos. Se cumple con el objetivo planteado de crear un sistema de clasificación de gestos que retorne resultados en un tiempo menor a 300 milisegundos.

3.2 Control de la posición angular del brazo robótico

Los controladores PID implementados deben asegurar un error en estado estable menor a 5° . Para validar su funcionamiento, se realizaron pruebas para diferentes entradas paso. Las entradas paso son referencias de posición angular de cada grado de libertad. Estas entradas paso se eligieron basados en los límites establecidos para cada grado de libertad (ver sección 0). En las siguientes figuras se presenta la respuesta obtenida para cada grado de libertad de un caso de prueba. Se incluye una tabla en la que se presenta la posición angular final θ_{ss} y el porcentaje de error en estado estable e_{ss} para los casos de prueba. Las curvas con las respuestas para todos los casos de prueba se encuentran en el ANEXO V. RUTINA DE PRUEBAS.

Es importante mencionar aquí que las señales de respuesta presentadas a continuación, así como también la señal de realimentación para los controladores utilizan la medición del tacómetro de los motores largos (M_1 , M_2 y M_3) del kit Lego Mindstorms. Por lo que, la medición de posición angular se obtiene al eje de giro de cada motor, y no al eje de giro de la articulación. Se calculan las posiciones angulares al eje de giro de cada articulación

utilizando la relación de engranes implementada en cada grado de libertad. En este caso, se considera que la relación de engranes es ideal, y que no existe pérdidas de potencia mecánica en la transmisión del movimiento.

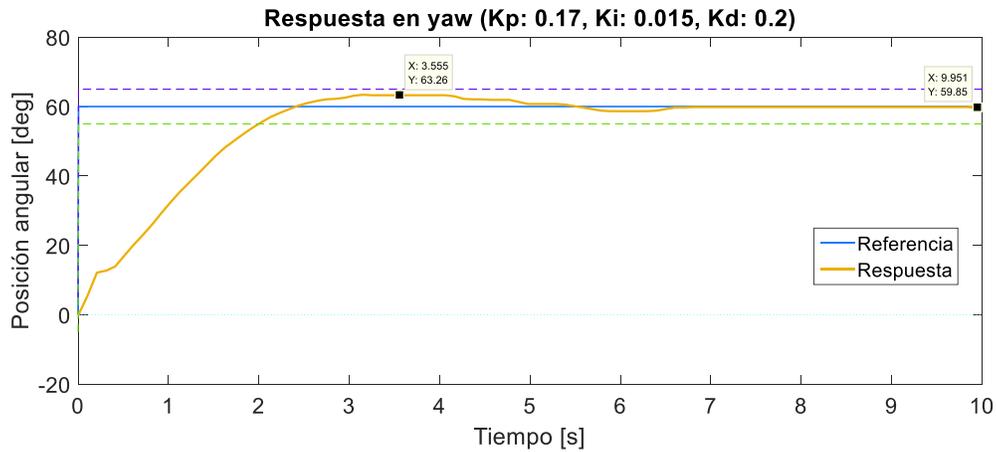


Figura 3.4 Respuesta del sistema del primer grado de libertad para una entrada paso de +60 grados (*yaw*)

Tabla 3.2 Resultados del controlador PID sobre el primer grado de libertad

Referencia [deg]	θ_{ss} [deg]	$\theta_{ss} - \theta_{ref}$ [deg]	e_{ss} [%]
-30	-30.52	-0.52	1.73
-60	-60.59	-0.59	0.983
+30	29.93	-0.07	-0.23
+60	59.85	-0.15	-0.25

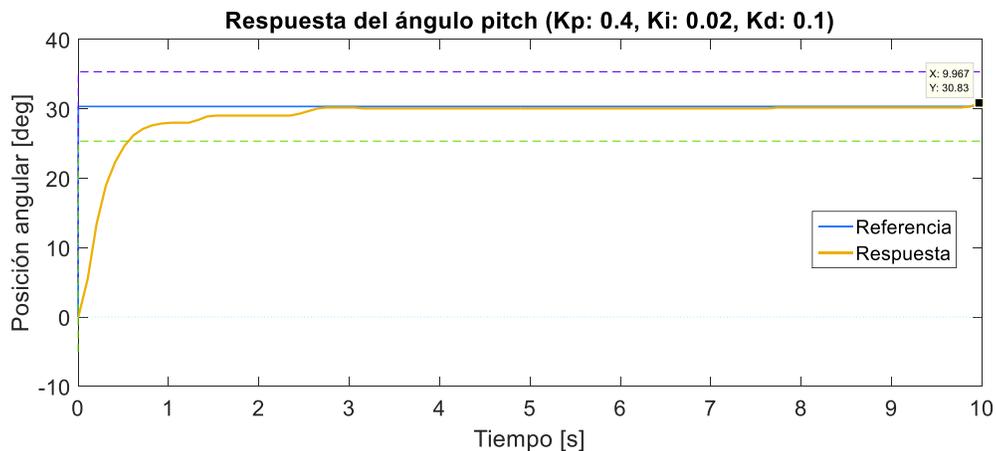


Figura 3.5 Respuesta del sistema del segundo grado de libertad para una entrada paso de +30 grados (*pitch*)

Tabla 3.3 Resultados del controlador PID sobre el segundo grado de libertad

Referencia [deg]	θ_{ss} [deg]	$\theta_{ss} - \theta_{ref}$ [deg]	e_{ss} [%]
-30	-30.19	-0.19	0.63
+30	30.83	0.83	2.76
+60	60.37	0.37	0.61

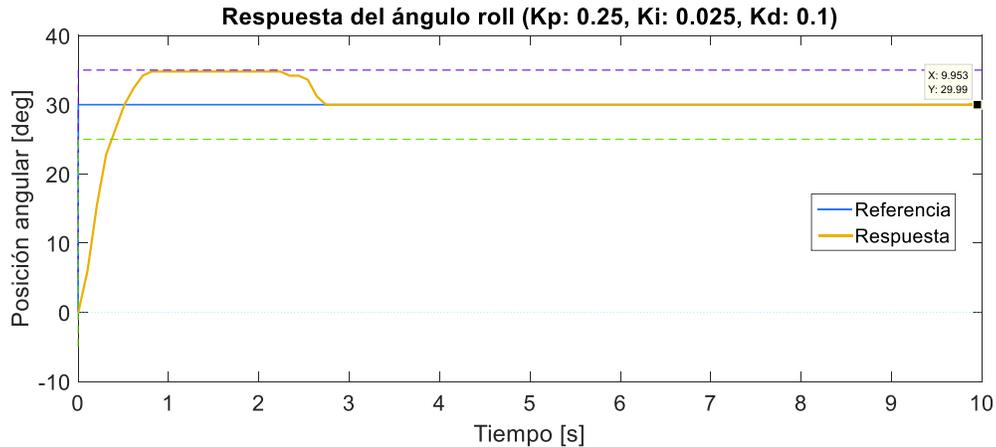


Figura 3.6 Respuesta del sistema del tercer grado de libertad para una entrada paso de +30 grados (*roll*)

Tabla 3.4 Resultados del controlador PID sobre el tercer grado de libertad

Referencia [deg]	θ_{ss} [deg]	$\theta_{ss} - \theta_{ref}$ [deg]	e_{ss} [%]
30	29.99	-0.01	-0.03
60	61.79	1.79	2.98
90	89.38	-0.62	-0.69

En las Figura 3.4, Figura 3.5 y Figura 3.6 se observa que para ningún grado de libertad se tiene un error mayor a 5° en la posición angular final. En estas figuras se incluyen líneas entrecortadas que muestran las bandas $\pm 5^\circ$ de la posición angular de referencia. De los casos presentados, se evidencia además que, en ningún caso, la respuesta del sistema tiene un sobrepico mayor a 5°. En las Tabla 3.2 Tabla 3.3 y Tabla 3.4 se presenta las posiciones angulares finales para todos los casos de prueba. De todos los casos de prueba, la referencia con mayor porcentaje de error se da en el tercer grado de libertad (ángulo *roll*) con un 2.98%. Este error, a pesar de ser el máximo obtenido, corresponde a una diferencia angular de 1.79°, menor a los 5° establecidos en el alcance de este proyecto.

4. CONCLUSIONES Y RECOMENDACIONES

4.1 Conclusiones

De las pruebas realizadas para medir la exactitud de clasificación y el tiempo de respuesta del sistema, así como también de las pruebas realizadas para diferentes entradas paso de los controladores PID implementados, se concluye que el presente proyecto cumple con los objetivos establecidos. Con respecto al clasificador de gestos, se puede concluir que: los parámetros escogidos al ser utilizados conjuntamente forman un sistema de clasificación que cumple con tener una exactitud de clasificación mayor al 85%, y un tiempo de respuesta menor a 300 milisegundos (ver sección 3.1). Mas, no es posible concluir que cada uno de estos parámetros haya sido escogido de la mejor manera, pues al ser un sistema en cascada, posibles errores, o bajos desempeños, pueden estar camuflados por el resultado global; solamente es posible realizar un análisis del efecto que tiene cada uno de estos en la respuesta del sistema. Los resultados globales de las pruebas indican que:

- El haber utilizado el Myo Armband para la adquisición de las señales EMG fue una buena elección para este proyecto de titulación. Pues su batería de larga duración permitió jornadas de prueba de varias horas. Además, para el sistema de clasificación implementado, la frecuencia de muestreo del Myo Armband (200Hz, ver sección 0) fue suficiente.
- Para que la ventana de tiempo contenga toda la señal EMG correspondiente a un gesto, la ventana debe durar alrededor de 2 segundos. Este parámetro se estableció a partir del análisis de las curvas amplitud-tiempo de los gestos de entrenamiento (ver sección 0 y 2.3) de tal manera que no fuese recortada la señal EMG durante la grabación. Sin embargo, aunque la duración de la ventana es tiempo suficiente para contener a cualquiera de los gestos definidos, fue necesario incluir en la interfaz de usuario la opción de repetir una grabación de algún gesto. Con esta última consideración, y con el tiempo de duración de la ventana establecido, se asegura que el sistema adquiera correctamente la señal EMG para la clasificación.
- El haber incluido la categoría adicional del gesto “reposo” (ver sección 0) a los gestos de la rutina de entrenamiento funcionó como fondo para la clasificación de las señales EMG, a pesar de no ser una buena representación del universo de gestos

posibles. Con esta inclusión se evita que varios gestos diferentes a los elegidos sean reconocidos por el sistema.

- El algoritmo DTW se utilizó como medida de distancia entre las señales EMG de los gestos. Esta comparación funcionó incluso entre gestos de diferente duración y desalineados en el tiempo. Se considera, sin embargo, que esta funcionalidad del algoritmo DTW se da únicamente después de aplicar el preprocesamiento a la señal EMG (filtro pasa bajos sobre la señal rectificada, sección 0); y que, para señales EMG sin procesar, se obtienen peores resultados de alineamiento.
- El criterio utilizado para determinar el valor de k (ver sección 0) es una regla empírica utilizada debido a que el número de muestras de entrenamiento, N , es finito. Basados en los resultados de clasificación, se concluye que el criterio elegido para establecer el número de vecinos, k , fue correcto.
- De las pruebas realizadas se concluye que el algoritmo implementado tarda menos de 300 milisegundos en la clasificación. No obstante, el resultado de tiempos de clasificación no es constante; depende de la carga computacional y varía de computador a computador. Se reportó las características técnicas de la computadora en la que se realizaron pruebas; por lo que se espera que en computadores de las mismas características o superiores el tiempo de respuesta cumplirá con el requisito establecido.
- La etapa de posprocesamiento del gesto resultante (ver sección 0) se implementó para eliminar las discontinuidades en la predicción del sistema de clasificación. Al incluir esta etapa se redujo el número de comandos de activación erróneos para el brazo robótico. Por lo que podemos concluir que la etapa de posprocesamiento mejora el sistema de clasificación al utilizarlo para el comando del brazo robótico.

Para confirmar que el presente proyecto cumple con los objetivos establecidos, con respecto al comando del brazo robótico de 3 grados de libertad, y a los controladores PID implementados, se realizaron las pruebas descritas en la sección 3.2. De tales pruebas se concluye que.

- La señal de orientación adquirida a través del Myo Armband está referenciada con respecto a los ejes magnéticos de la Tierra. Esta señal en 3 ejes coordenados fue utilizada como referencia para la teleoperación del brazo robótico; y da buenos re-

sultados siempre y cuando se mantenga el brazo estirado (ver sección 0). Esto debido a que, utilizando solo un Myo Armband, no es posible medir la orientación relativa del antebrazo con respecto al brazo o al torso del usuario.

- Los controladores PID cumplen con tener un error en estado estable menor a 5 grados de la posición angular. Para calcular la posición angular al eje de giro de cada articulación se utiliza la medición del tacómetro de los motores del kit Lego Mindstorms y la relación de trabajo ideal de los engranes utilizados. Por lo que, para el presente trabajo, no se considera el error de posición que se produce por la holgura entre los dientes de los engranes.

4.2 Recomendaciones

Para aquellos que continúen en la investigación, ya sea en el área de la clasificación de gestos o en el área de la robótica, se presenta las siguientes recomendaciones a tener en cuenta:

- Se recomienda utilizar el Myo Armband en el desarrollo de sistemas de clasificación de gestos. Esto debido principalmente a su bajo costo, portabilidad y facilidad de uso; pues ofrece al usuario las señales EMG de manera inalámbrica ya acondicionadas. Además, teniendo en cuenta que el Myo Armband posee una unidad de medición inercial, es posible incorporar a las señales de orientación.
- La ventana de tiempo de 2 segundos escogida para el entrenamiento aseguró no tener recortes de la señal EMG. Sin embargo, la señal adquirida incluye segmentos de señal no correspondientes al gesto realizado. Esto puede ocasionar problemas en la clasificación, y ralentiza al sistema. Se recomienda, implementar un algoritmo de segmentación para recortar las señales EMG del entrenamiento a los puntos correspondientes solamente al gesto realizado, de esta manera se podría acelerar la respuesta del sistema y conseguir mejores tasas de clasificación.
- Se recomienda utilizar el algoritmo de alineamiento temporal dinámico, DTW, para la comparación de series de tiempo. Pues este algoritmo es robusto ante diferencias de duración y desalineamientos en tiempo. Sin embargo, se debe tener en cuenta que computacionalmente es muy costoso, y que su complejidad aumenta al cuadrado del tamaño de cada vector.
- El clasificador kNN usado en este trabajo considera los 5 vecinos más cercanos para la clasificación. Este parámetro se estableció heurísticamente considerando el criterio de consistencia, ya que no puede ser mayor al número de repeticiones del

entrenamiento. Se recomienda también probar otros clasificadores y realizar comparaciones con el algoritmo propuesto en este trabajo.

- El sistema implementado de clasificación de gestos es usuario dependiente, pues cada usuario que desee utilizar el sistema debe realizar la rutina de entrenamiento. Esto debido a las diferencias fisiológicas entre persona y persona, y también, por que el Myo Armband no necesariamente se encuentra en la misma posición en cada ocasión de uso. Se recomienda realizar comparaciones al utilizar otros sensores para la adquisición de la señal EMG.
- Se recomienda utilizar el kit Lego Mindstorms EV3 para la realización de proyectos en robótica, pues este kit ofrece varias ventajas. Los actuadores, por ejemplo, están listos para usarse, y ofrecen mediciones del torque y la velocidad. Además, el kit Lego Mindstorms tiene una gran variedad de sensores, lo que permite desarrollar un gran número de aplicaciones. También, es una ventaja del kit Lego Mindstorms que exista varias interfaces desde las que se puede programar al ladrillo inteligente EV3, (Matlab, Labview, etc.). Por lo que el Lego Mindstorms es recomendable para proyectos que requieran una fácil implementación.

5. REFERENCIAS BIBLIOGRÁFICAS

- [1] J. G. Abreu, J. M. Teixeira, L. S. Figueiredo y V. Teichrieb, «Evaluating Sign Language Recognition Using the Myo Armband,» *2016 XVIII Symposium on Virtual and Augmented Reality*, pp. 65-66, 2016.
- [2] D. C. García Cortés, Reconocimiento de Gestos de Manos como Mecanismo de Interacción Humano-Computador, Bogotá: Universidad Nacional de Colombia, 2014.
- [3] L. Estrada, Diseño e implementación de un prototipo para la traducción de lenguaje de señas mediante la utilización de un guante sensorizado, Quito: EPN, 2016.
- [4] M. Reaz, M. Hussain y F. Mohd-Yasin, «Techniques of EMG signal analysis: detection, processing, classification and applications,» *Biological Procedures Online*, vol. 8, nº 1, pp. 11-35, 2006.
- [5] M. R. Ahsan, M. I. Ibrahimy y O. O. Khalifa, «Electromyography (EMG) signal based hand gesture recognition using artificial neural network (ANN),» de *4th International Conference on Mechatronics (ICOM)*, Kuala Lumpur, 2011.
- [6] J. Chu, I. Moon y M. Mun, «A real-time EMG Patern Recognition System based on linear-nonlinear feature projection for a multifunction myoelectric hand,» *IEEE Transactions on Biomedical Engineering*, vol. 53, nº 11, p. 2233, 2006.
- [7] L. Weiss, J. Weiss y J. Silver, *Easy EMG*, 2nd ed., Elsevier, 2016.
- [8] M. Mordhorst, T. Heidlauf y O. Röhrle, «Mathematically modelling surface EMG signals,» *PAMM*, vol. 14, nº 1, pp. 123-124, 2014.
- [9] C. D.Katsis, T. P. Exarchos, C. Papaloukas, Y. GOletsis, D. I. Fotiadis y I. Sarmas, «A two-stage method for MUAP classification based on EMG decomposition,» *Computers in Biology and Medicine*, vol. 37, nº 9, pp. 1232-1240, 2007.
- [10] C. J. De Luca, «Imaging the Behavior of Motor Units,» Delsys Inc., Boston, 2008.
- [11] M. Gallegos y J. Simbaña, «Control Del Movimiento De Una Pinza Robótica Por Medio De La Actividad Neuronal Del Antebrazo,» Quito, EPN, 2006.
- [12] M. Hidalgo y A. S. G Tene, «Fuzy control of a robotic arm using EMG signals,» de *2005 International Conference on Industrial Electronics and Control Applications (FIEE)*, Quito, EPN, 2005.
- [13] L. J. Myers, M. Lowery, C. Vaughan, A. C. Gibson, Y. Harley y R. Sreenivasan, «Rectification and non-linear pre-processing of EMG signals for cortico-muscular analysis,» *Journal of Neuroscience Methods*, vol. 124, nº 2, pp. 157-165, 2003.
- [14] P. Bonato, T. D'Alessio y M. Knaflitz, «A Statistical Method for the Measurement of Muscle Activation Intervals from Surface Myoelectric Signal During Gait,» *IEEE Transactions on Biomedical Engineering*, vol. 45, nº 3, pp. 287-299, 1998.

- [15] L. Xu y A. Adler, «An improved method for muscle activation detection during gait,» *Canadian Conference on Electrical and Computer Engineering*, pp. 357-260, 2004.
- [16] F. Pérez Ramírez, *Introducción a las series de tiempo. Métodos paramétricos*, Universidad De Medellín, 2007.
- [17] Thalmic Labs Inc., «Unveiling the Final Design of the Myo™ Armband,» 10 junio 2014. [En línea]. Available: <http://blog.thalmic.com/unveiling-final-design-myo-armband/>.
- [18] Thalmic Labs Inc., «Myo Gesture Control Armband,» Myo.com, 2016. [En línea]. Available: <https://www.myo.com/techspecs>.
- [19] M. Rouse, «DEFINITION: Gesture Recognition,» WhatIs, [En línea]. Available: <http://whatIs.techtarget.com/definition/gesture-recognition>.
- [20] G. Baratoff y D. Searles, «Gesture Recognition,» *The Encyclopedia of Virtual Environments*, 1993. [En línea]. Available: http://www.hitl.washington.edu/projects/knowledge_base/virtual-worlds/EVE/I.D.2.b.GestureRecognition.html.
- [21] Y. Huang, W. Guo, J. Liu, J. He, H. Xia, X. Sheng, H. Wang, X. Feng y P. B. Shull, «Preliminary Testing of a Hand Gesture Recognition Wristband Based on EMG and Inertial Sensor Fusion,» de *Intelligent Robotics and Applications. ICIRA 2015. Lecture Notes in Computer Science*, vol. 9244, Springer, Cham, 2015.
- [22] National Instruments Corporation, «¿Qué es Adquisición de Datos?,» [En línea]. Available: <http://www.ni.com/data-acquisition/what-is/esa/>.
- [23] U. Akthar y M. Hassan, «Big Data Mining Based on Computational Intelligence and Fuzzy Clustering,» de *Handbook of Research on Trends and Future Directions in Big Data and Web Intelligence*, IG Global, 2015, pp. 130-148.
- [24] Universidad Autónoma de Barcelona, «Clasificación de imágenes: ¿cómo reconocer el contenido de una imagen?,» Coursera, 2017. [En línea]. Available: <https://www.coursera.org/learn/clasificacion-imagenes/home/welcome>.
- [25] Universidad de Stanford, «Aprendizaje Automático,» Coursera, 2017. [En línea]. Available: <https://www.coursera.org/learn/machine-learning/home/info>.
- [26] R. Scherer y R. Rao, «Non-manual Control Devices: Direct Brain-Computer Interaction,» de *Handbook of Research on Personal Autonomy Technologies and Disability Informatics*, IGI Global, 2011, pp. 233-250.
- [27] D. Rodriguez, A. Pryanrinska y X. Zhang, «A framework for hand gesture recognition based on accelerometer and emg sensors,» *IEEE Transactions on Systems, Man and Cybernetics*, vol. 41, nº 6, pp. 1064-1076, 2011.
- [28] S. Ameer, B. Khalifa y M. S. Bouhlel, «A comprehensive leap motion database for hand gesture recognition,» de *2016 7th International Conference on Sciences of*

Electronics, Technologies of Information and Telecommunications (SETIT), Hammamet, 2016.

- [29] Z. Arief, I. A. Sulistijono y R. A. Ardiansyah, «Comparison of five time series EMG features extractions using Myo Armband,» de *Electronics Symposium (IES) 2015 International*, Surabaya, 2015.
- [30] Encyclopædia Britannica Inc., «Pattern Recognition,» [En línea]. Available: <https://www.britannica.com/technology/pattern-recognition-computer-science>.
- [31] J. Brownlee, «A Tour of Machine Learning Algorithms,» Machine Learning Algorithms, noviembre 2015. [En línea]. Available: <http://machinelearningmastery.com/a-tour-of-machine-learning-algorithms/>.
- [32] Thalmic Labs Inc., «Over 100 Applications and Connectors for Myo,» Myo Market, 2017. [En línea]. Available: <https://market.myo.com/>.
- [33] Y. Xu, C. Yang, P. Liang, L. Zhao y Z. Li, «Development of a Hybrid Motion Capture Method Using MYO Armband with Application to Teleoperation,» de *2016 IEEE International Conference on Mechatronics and Automation*, Harbin, 2016, pp. 1179-1184.
- [34] Thalmic Labs Inc., «Developer: Getting Started,» [En línea]. Available: https://developer.thalmic.com/docs/api_reference/platform/getting-started.html.
- [35] Thalmic Labs Inc., «Myo Developer: Downloads,» [En línea]. Available: <https://developer.thalmic.com/downloads>.
- [36] M. Tomaszewski, «MyoMex,» 13 marzo 2016. [En línea]. Available: <https://github.com/mark-toma/MyoMex>.
- [37] Thalmic Labs Inc., «How to wear the Myo armband,» Myo Support, [En línea]. Available: <https://support.getmyo.com/hc/en-us/articles/201169525-How-to-wear-the-Myo-aramband>.
- [38] A. V. Oppenheim y R. W. Schaffer, «Digital Filter Design Techniques,» de *Digital Signal Processing*, Prentice Hall, 1era edición, pp. 211-218.
- [39] R. C. González y R. E. Woods, «Filtering in the Frequency Domain,» de *Digital Image Processing*, Pearson, 3era edición, pp. 273-276.
- [40] H. Sakoe y C. S., «Dynamic programming algorithm optimization for spoken word recognition,» *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 26, nº 1, pp. 43-49, 1978.
- [41] M. Müller, «Dynamic Time Warping,» de *Information Retrieval for Music and Motion*, Springer, 2007, pp. 71-84.
- [42] E. Tsiorkova, «Dynamic Time Warping Algorithm for Gene Expression Time Series,» Universiteit Gent, [En línea]. Available: <https://www.psb.ugent.be/cbd/papers/gentxwarper/DTWalgorithm.htm>.

- [43] Macquarie University, «Dynamic Time Warping,» Matching Patterns in Time, [En línea]. Available: <http://web.science.mq.edu.au/~cassidy/comp449/html/ch11s02.html>.
- [44] C. A. Ratanamahatana y E. Keogh, «Making Time-series Classification More Accurate Using Learned Constraints,» *SIAM International Conference on Data Mining*, 2004.
- [45] C. Bishop, «Nonparametric Methods,» de *Pattern Recognition and Machine Learning*, Springer, 2006, pp. 120-127.
- [46] O. Sutton, «The k Nearest Neighbours Algorithm,» Universidad de Leicester, 2012. [En línea]. Available: http://www.math.le.ac.uk/people/ag153/homepage/KNN/OliverKNN_Talk.pdf.
- [47] J. Béjar, «K-nearest neighbours,» Universitat Politècnica de Catalunya, 2012. [En línea]. Available: <http://www.lsi.upc.edu/~bejar/apren/docum/trans/03d-algind-knn-eng.pdf>.
- [48] S. Thirumuruganathan, «KNN Introduction,» de *A Detailed Introduction to K-Nearest Neighbor (KNN) Algorithm*, 2010.
- [49] S. Sayad, «K Nearest Neighbors,» de *Classification*, 2001.
- [50] L. Devroye, L. Györfi y G. Lugosi, «A Probabilistic Theory of Pattern Recognition,» *Applications of Mathematics. Stochastic Modelling and Applied Probability*, vol. 31, pp. 61-81, 1991.
- [51] Y. Dong, «Universal Consistency of the k-NN classifier,» Universidad de Ottawa, 2014. [En línea]. Available: <http://mysite.science.uottawa.ca/shatko/SummerSeminar/OMC%20Talk.pdf>.
- [52] K. Dowling, «What is robotics?,» Carnegie Mellon University, 1995. [En línea]. Available: <https://www.cs.cmu.edu/~chuck/robotpg/robofaq/1.html>.
- [53] A. Barrientos, P. L. C. Balaguer y R. Aracil, Fundamentos de Robótica, 2da Edición, Madrid: McGraw-Hill, 2007.
- [54] D. Benedettelli, The Lego® Mindstorms® EV3 Laboratory, Chelsea, Michigan: No Starch Press, 2013.
- [55] L. Valk, The Lego® Mindstorms® EV3 Discovery Book, Illinois: No Starch Press, 2014.
- [56] P. Hurbain, «LEGO® 9V Technic Motors compared characteristics,» LEGO technical information, 2012. [En línea]. Available: <http://www.philohome.com/motors/motorcomp.htm>.
- [57] W11114m, «Simplest EV3 Robot Claw/Gripper,» Electronics, 2015. [En línea]. Available: <http://www.instructables.com/id/Simplest-EV3-Robot-ClawGripper/>.
- [58] P. Corke, «Robotics, Vision and Control,» *Springer*, vol. 73, nº 2da edición, 2013.

- [59] K. Ogata, *Ingeniería de Control Moderna*, Madrid: 5ta edición, Pearson, 2010.
- [60] K. Ogata, *Sistemas de Control en Tiempo Discreto*, Ciudad de México: 2da edición, Prentice Hall, 1996.
- [61] K. J. Åström y R. M. Murray, «PID Control,» de *Control Systems Design*, 2002, pp. 216-251.
- [62] S. Manasa, S. Rani y M. Veda, «Position Control of a DC Motor Using PID Controller,» de *International Journal of Scientific Engineering and Applied Science (IJSEAS)*, 2015.
- [63] P. Bernhardt, «Magnetometer Data,» julio 2016. [En línea]. Available: <https://developer.thalmic.com/forums/topic/2645/>.
- [64] B. Mambrito y C. J. De Luca, «A technique for the detection, decomposition and analysis of the EMG signal,» *Electroencephalography and clinical Neurophysiology*, pp. 175-188, 1984.
- [65] C. Ratanamahatana y E. Keogh, «Everything you know about dynamic time warping is wrong,» *Third Workshop on Mining Temporal and Sequential Data*, 2004.

6. ANEXOS

ANEXO I. SEÑALES ELECTROMIOGRÁFICAS

Señales EMG puras

Se presenta de la Figura I.1 a la Figura I.5 las señales EMG puras de cada uno de los gestos del sistema de clasificación.

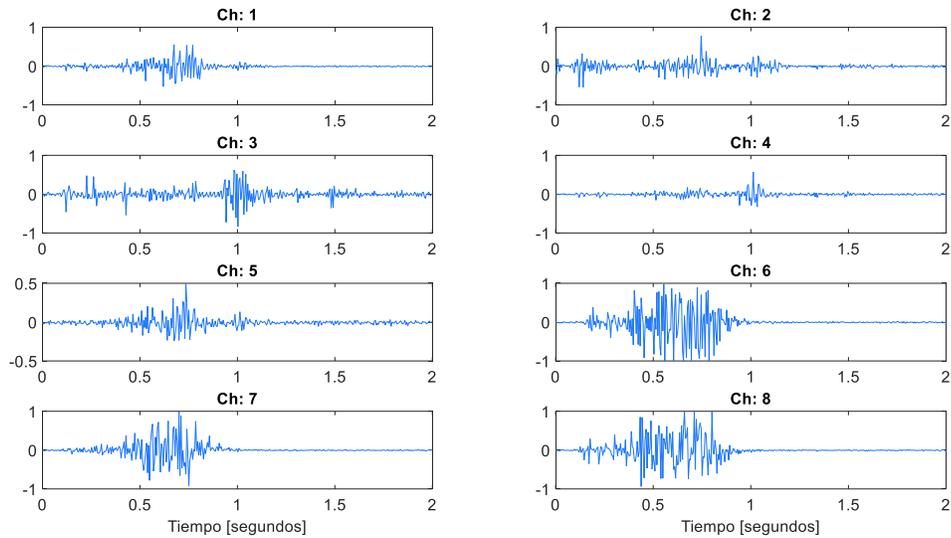


Figura I.1 Repetición del gesto mano en 90° hacia adentro (*Wave In*)

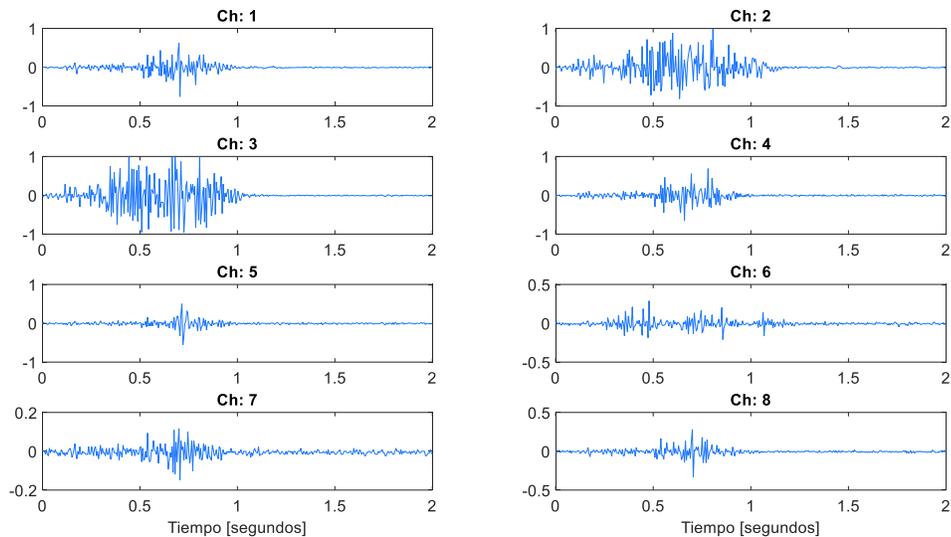


Figura I.2 Repetición del gesto mano en 90° hacia afuera (*Wave Out*)

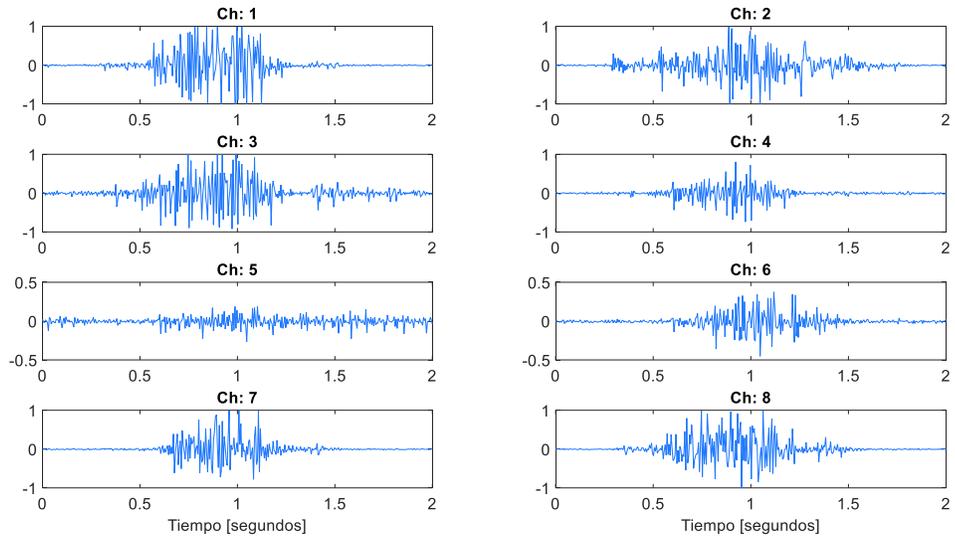


Figura I.3 Repetición del gesto puño (*Fist*)

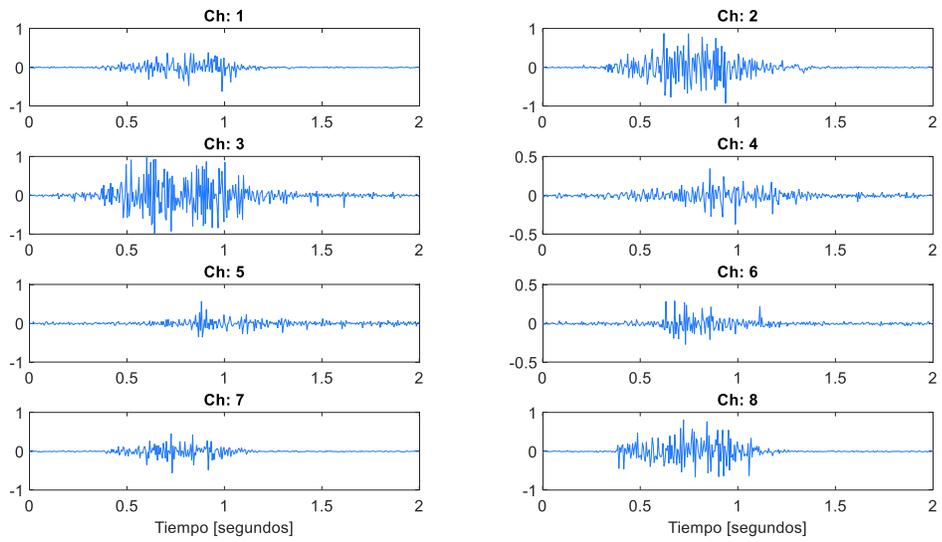


Figura I.4 Repetición del gesto mano abierta (*Open*)

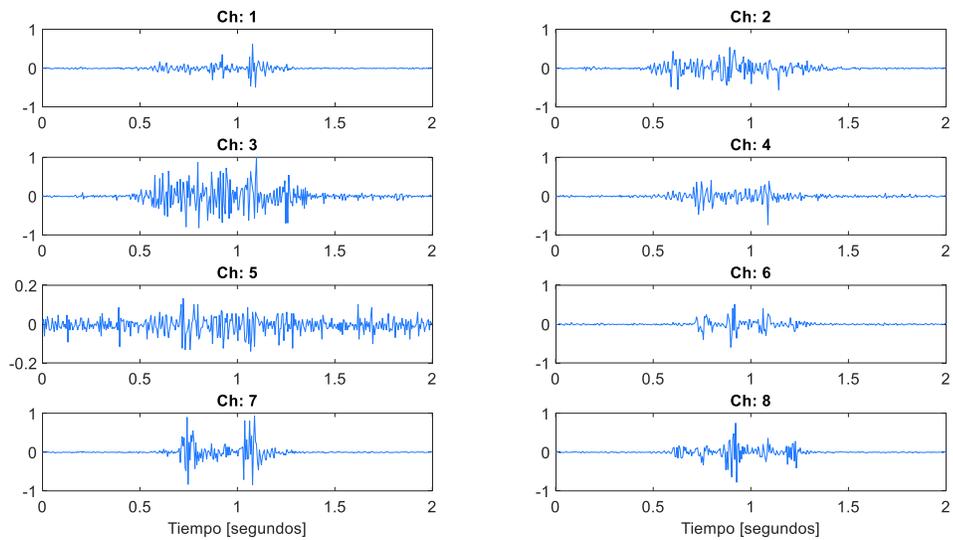


Figura I.5 Repetición del gesto pellizco entre dedo pulgar e índice (*Pinch*)

Señales EMG posprocesadas

Se realizaron grabaciones de cada gesto por 10 segundos. En esos 10 segundos se realizan tantas repeticiones naturales como fueron posibles. A estas repeticiones se aplicó la rectificación y el filtrado. De la Figura a la Figura se presenta a cada uno de los gestos repetidos entre 7 y 9 veces. Para facilitar la visualización, se muestra dos canales del Myo Armband en cada eje.

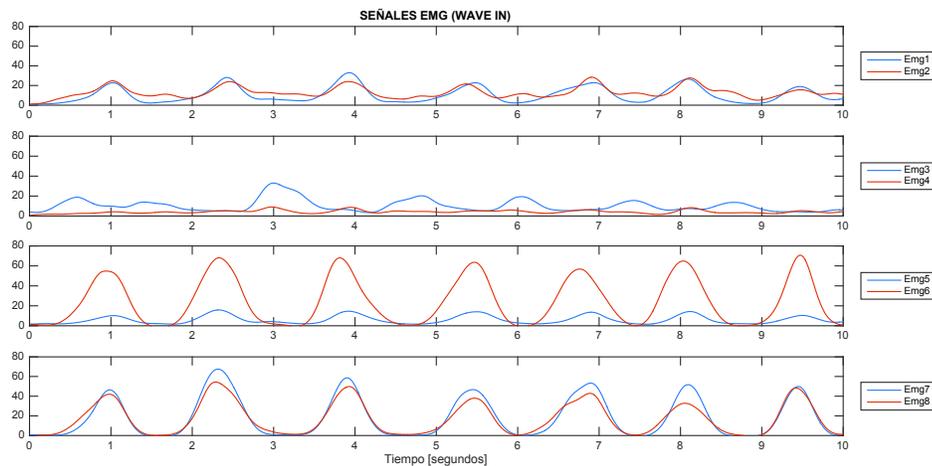


Figura I.6 Repeticiones del gesto *wave in*

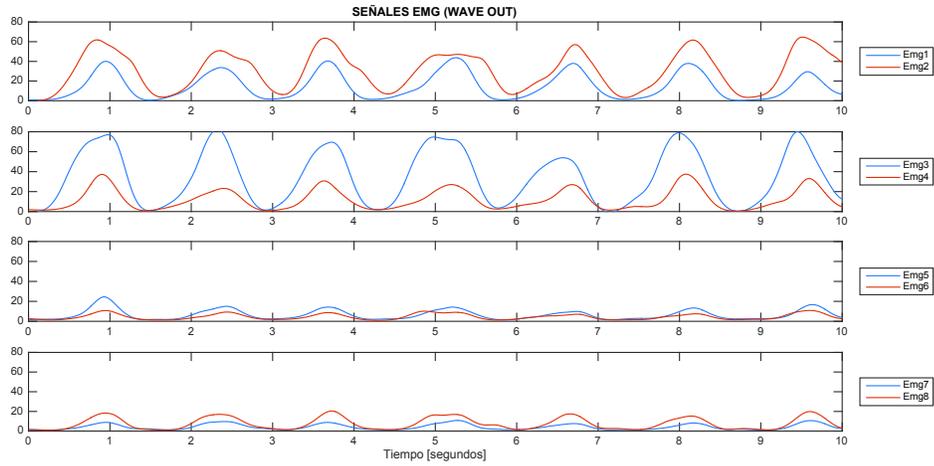


Figura I.7 Repeticiones del gesto *wave out*

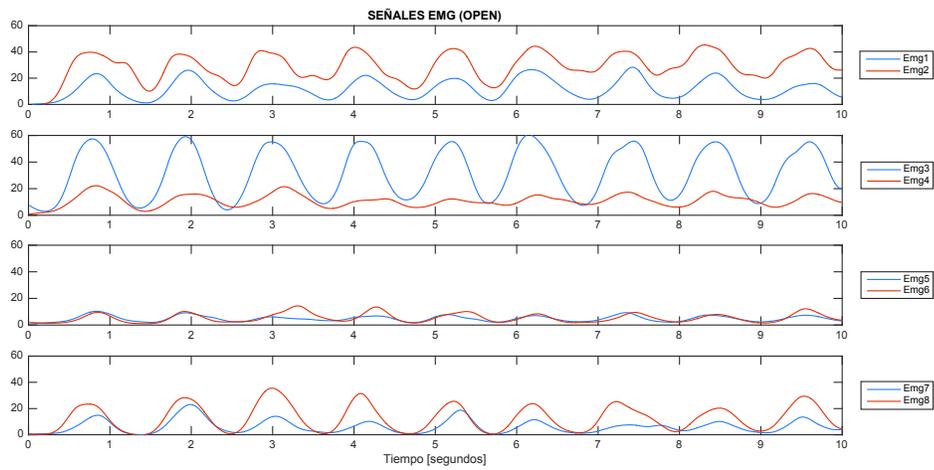


Figura I.8 Repeticiones del gesto *open*

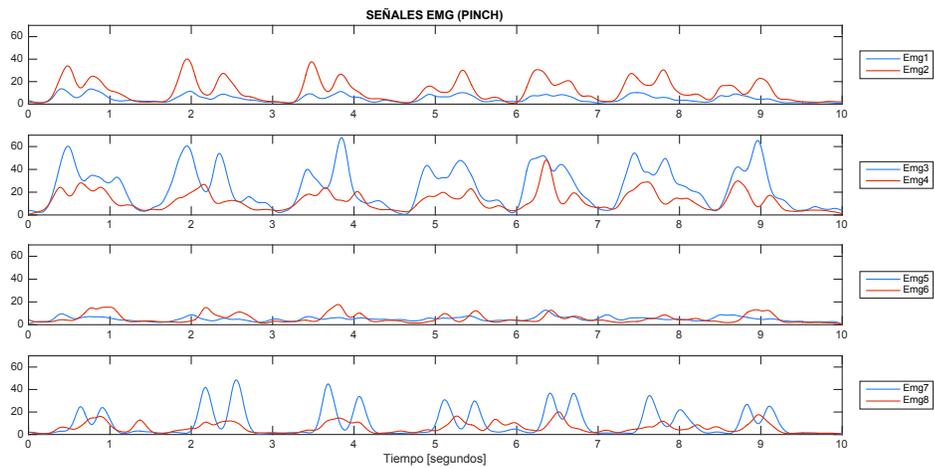


Figura I.9 Repeticiones del gesto *pinch*

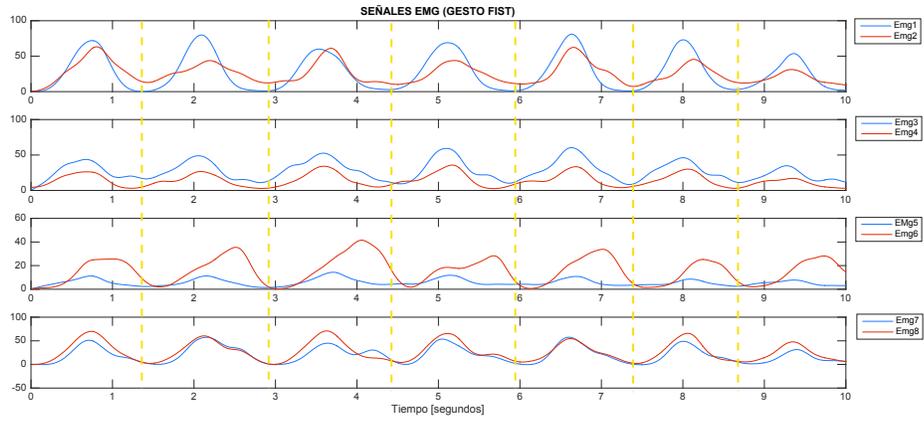
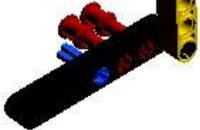
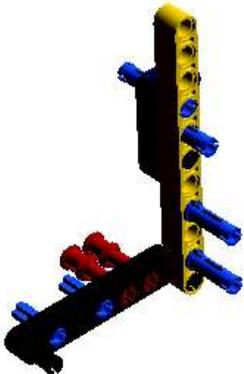
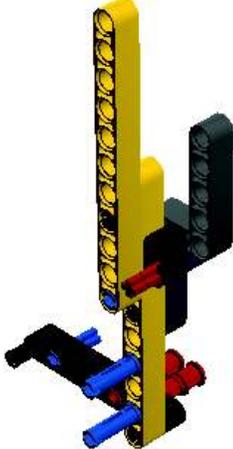


Figura I.10 Repeticiones del gesto *fist*

ANEXO II. INSTRUCCIONES DE ARMADO

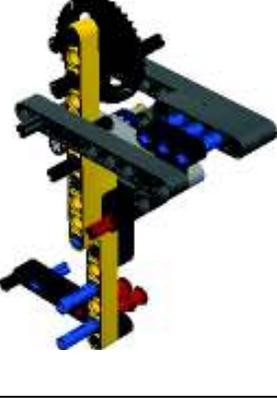
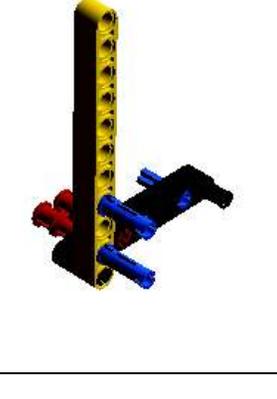
A continuación, se presenta las instrucciones para el armado del brazo robótico utilizando el kit Lego Mindstorms EV3. Estas instrucciones fueron generadas por el software Lego Digital Designer. Se divide este ANEXO en cuatro partes: construcción de la base, primer eslabón, segundo eslabón y pinza robótica.

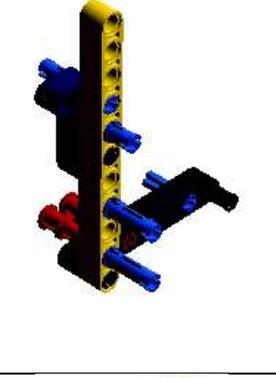
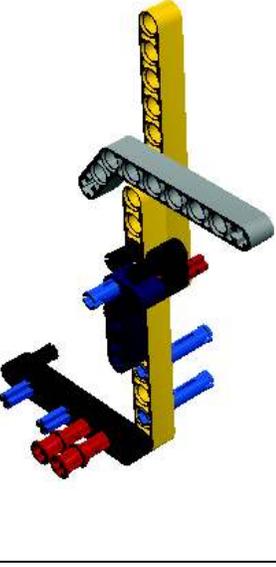
TABLA II.1 Proceso de armado del primer eslabón del brazo robótico

Piezas		Vista
1x		
1x		
1x		
1x		
1x		
2x		
4x		
1x		
1x		
3x		
1x		
1x		
1x		
1x		

1x		
4x		
1x		

1x		
2x		
1x		

1x		
1x		
2x		
3x		
1x		
1x		
2x		
2x		
2x		

1x		
1x		
2x		
1x		
2x		
1x		
1x		
1x		
1x		

1x		
3x		
1x		
2x		
1x		

1x		
2x		
1x		
2x		
1x		
2x		
1x		
1x		

1x		
4x		
1x		
2x		
1x		
1x		
2x		
2x		

1x		
4x		
1x		
2x		
2x		
1x		
1x		
3x		
1x		

2x		
1x		
2x		
3x		
1x		
2x		
2x		
2x		

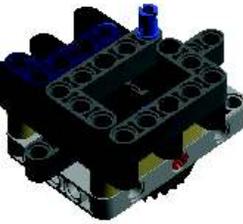
2x		
1x		
2x		
1x		
2x		
3x		
1x		

1x		
1x		
1x		
2x		
1x		
2x		
2x		
1x		

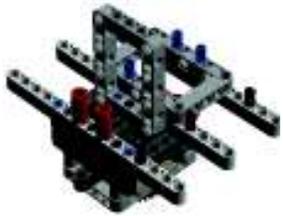
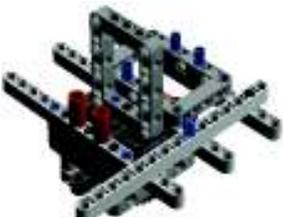
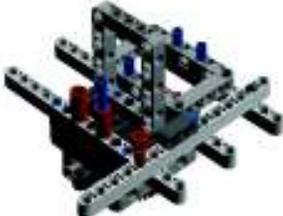
1x		
1x		
2x		
2x		
1x		
1x		
1x		
2x		
4x		
2x		
1x		
4x		

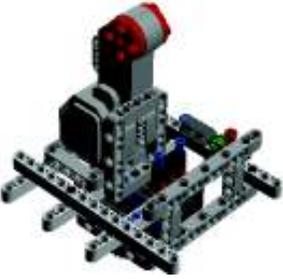
1x		
1x		
2x		
1x		
1x		
1x		
1x		
1x		
1x		

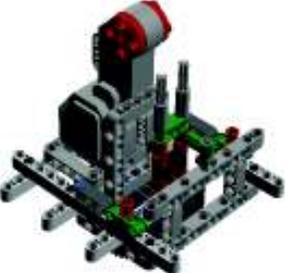
1x		
1x		
1x		
1x		
5x		
2x		
1x		
2x		
1x		
1x		
1x		
1x		

1x		
3x		
2x		
3x		
1x		
2x		
1x		
2x		
1x		
1x		
1x		
1x		
1x		
1x		

6x		
2x		
1x		
1x		
2x		
1x		
1x		
1x		
1x		
1x		
1x		

2x		
1x		
1x		
1x		
1x		
1x		
1x		
1x		
1x		
1x		
1x		
1x		

1x		
1x		
1x		
1x		
1x		
3x		
1x		
1x		
1x		
2x		
1x		

1x		
1x		
1x		
1x		
2x		
2x		

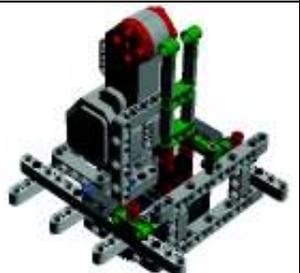
1x		
2x		
1x		
1x		
2x		
1x		

TABLA II.2 Proceso de armado del segundo eslabón del brazo robótico

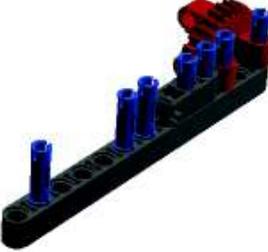
Piezas		Vista																							
1x																									
1x																									
1x																									
1x																									
1x																									
1x																									
2x																									
3x																									
1x																									
1x																									
1x																									
1x																									
1x																									
1x		<table border="1"> <tbody> <tr> <td>1x</td> <td></td> <td rowspan="11"></td> </tr> <tr> <td>1x</td> <td></td> </tr> <tr> <td>3x</td> <td></td> </tr> <tr> <td>1x</td> <td></td> </tr> <tr> <td>1x</td> <td></td> </tr> </tbody> </table>	1x			1x		3x		1x		1x													
1x																									
1x																									
1x																									
1x																									
1x																									
1x																									
1x																									
1x																									
3x																									
1x																									
1x																									

1x		
1x		
1x		
2x		
2x		
1x		
2x		
1x		
1x		
1x		
3x		
1x		

1x		
2x		
1x		
2x		
1x		

1x		
1x		
2x		
1x		
1x		
2x		
2x		
1x		
2x		
1x		
1x		
1x		
2x		

1x		
2x		
1x		
1x		
1x		
1x		
1x		
1x		
1x		
1x		
1x		
1x		
1x		

1x		
1x		
1x		
3x		
3x		
1x		

1x		
1x		
2x		
1x		
1x		

TABLA II.3 Proceso de armado de la base del brazo robótico

Piezas		Vista
1x		

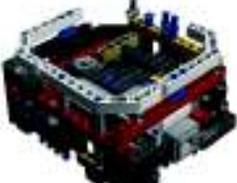
1x		
2x		
1x		
2x		
1x		
3x		

1x		
2x		
1x		

1x		
3x		
5x		
1x		

1x		
2x		
1x		
1x		
1x		
2x		
1x		

1x		
4x		
2x		
2x		
1x		
1x		
2x		
1x		

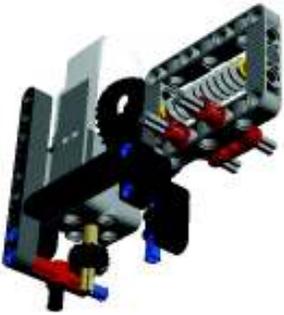
1x		
1x		
1x		
1x		
3x		
1x		
2x		
1x		
2x		
1x		
2x		
1x		
1x		

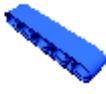
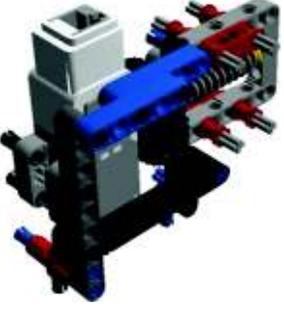
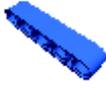
2x		
1x		
1x		
1x		
1x		
8		
7		
1x		
2x		
4x		

TABLA II.1 Proceso de armado de la pinza del brazo robótico

Piezas		Vista
1x		
1x		
2x		
1x		
2x		
1x		
1x		
1x		
1x		

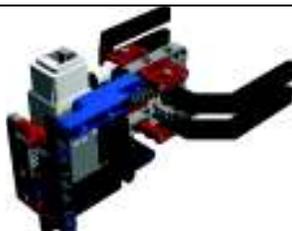
1x		
1x		
1x		
1x		
1x		
1x		
1x		
1x		
1x		
1x		

1x		
1x		
2x		
1x		
2x		
2x		
1x		
1x		
1x		
1x		

1x		
1x		
1x		
1x		
1x		
1x		
3x		
1x		
1x		
1x		

1x		

1x		

1x		
2x		
2x		
1x		
1x		
1x		

ANEXO III. MANUAL DE USUARIO

Este manual describe las características y requerimientos de la interfaz de usuario implementada.

Los requisitos previos para el sistema son:

- Myo Connect,
- Matlab R2015a,
- Myo SDK 0.9.0.

La conexión con el Myo Armband se realiza desde el Myo Connect. Para la conexión se debe abrir Armband Manager del menú del Myo Connect. Desde esta pantalla se puede conectar y desconectar el Myo Armband.



Figura III.1 Menú de Myo Connect

La comunicación con el brazo robótico implementado en Lego Mindstorms EV3 es vía Bluetooth. Para la conexión con el brazo robótico se debe establecer el vínculo entre el PC y el ladrillo inteligente EV3. En el menú Bluetooth de la pestaña de configuraciones del EV3, escoger la PC para realizar la conexión.



Figura III.2 Estableciendo conexión Bluetooth entre EV3 y computador

El sistema se ejecuta en simultáneo en Matlab y en el ladrillo inteligente EV3. En Matlab se debe correr el script "ejecutar.m". En el EV3 se debe ejecutar el programa "fullPID".

Como se explicó en la sección 2.5, la interfaz de usuario permite conectarse con el Myo Armband y el EV3, realizar la rutina de entrenamiento, visualizar las señales EMG y de orientación y ver los resultados obtenidos.

Para las establecer las conexiones se debe dar clic en “CONECTAR” del panel Conexión.



Figura III.3 Interfaz gráfica (panel de conexión)

ANEXO IV. DIAGRAMAS DE FLUJO

En este apartado se presenta los diagramas de flujo del sistema de clasificación de gestos implementado, así como también del mando a distancia del brazo robótico. El sistema de clasificación de gestos se ejecuta únicamente en Matlab 2015^a; mientras que el comando del brazo robótico incluye la comunicación entre Matlab y el ladrillo inteligente EV3.



Figura IV.1 Leyenda de los diagramas de flujo

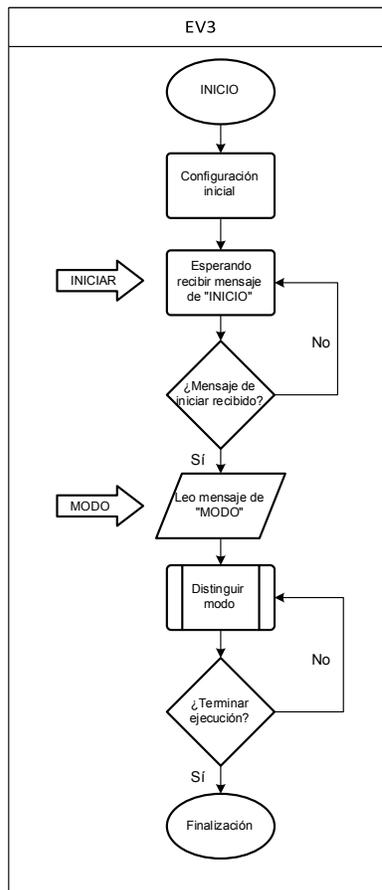


Figura IV.2 Diagrama de flujo general EV3

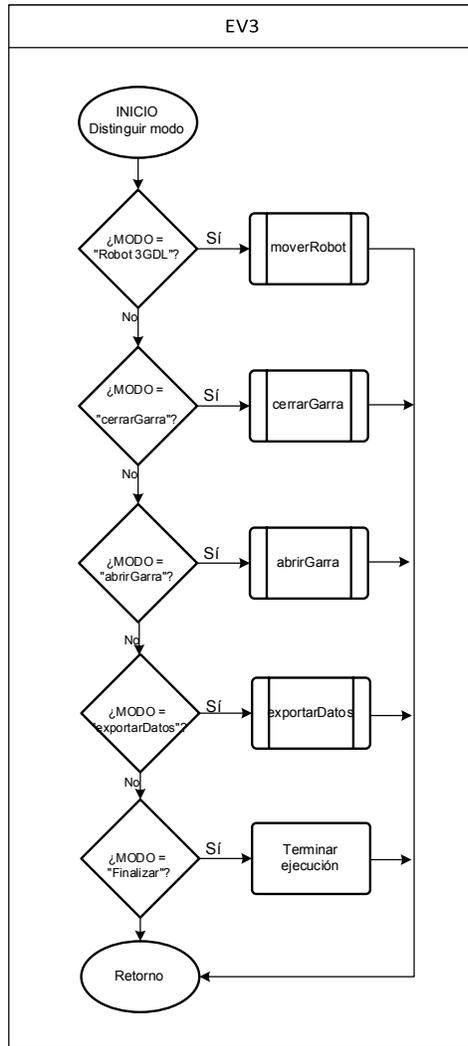


Figura IV.3 Modos de trabajo del ladrillo inteligente EV3

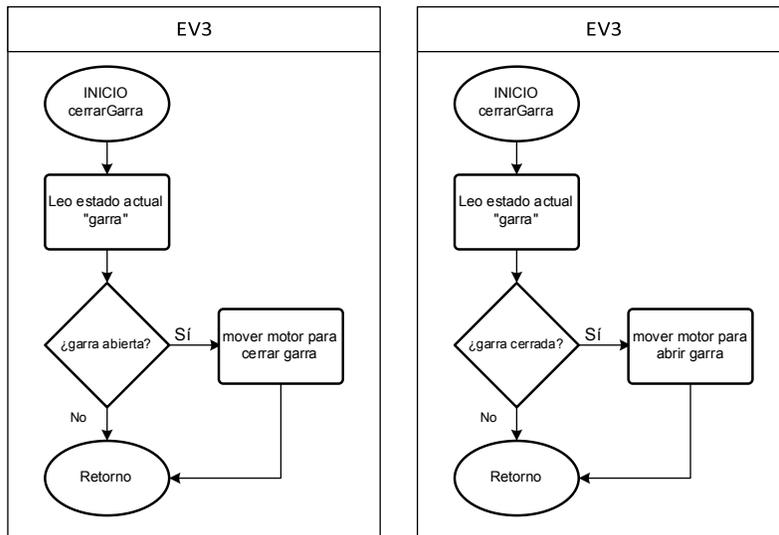


Figura IV.4 Rutinas para control de la pinza robótica

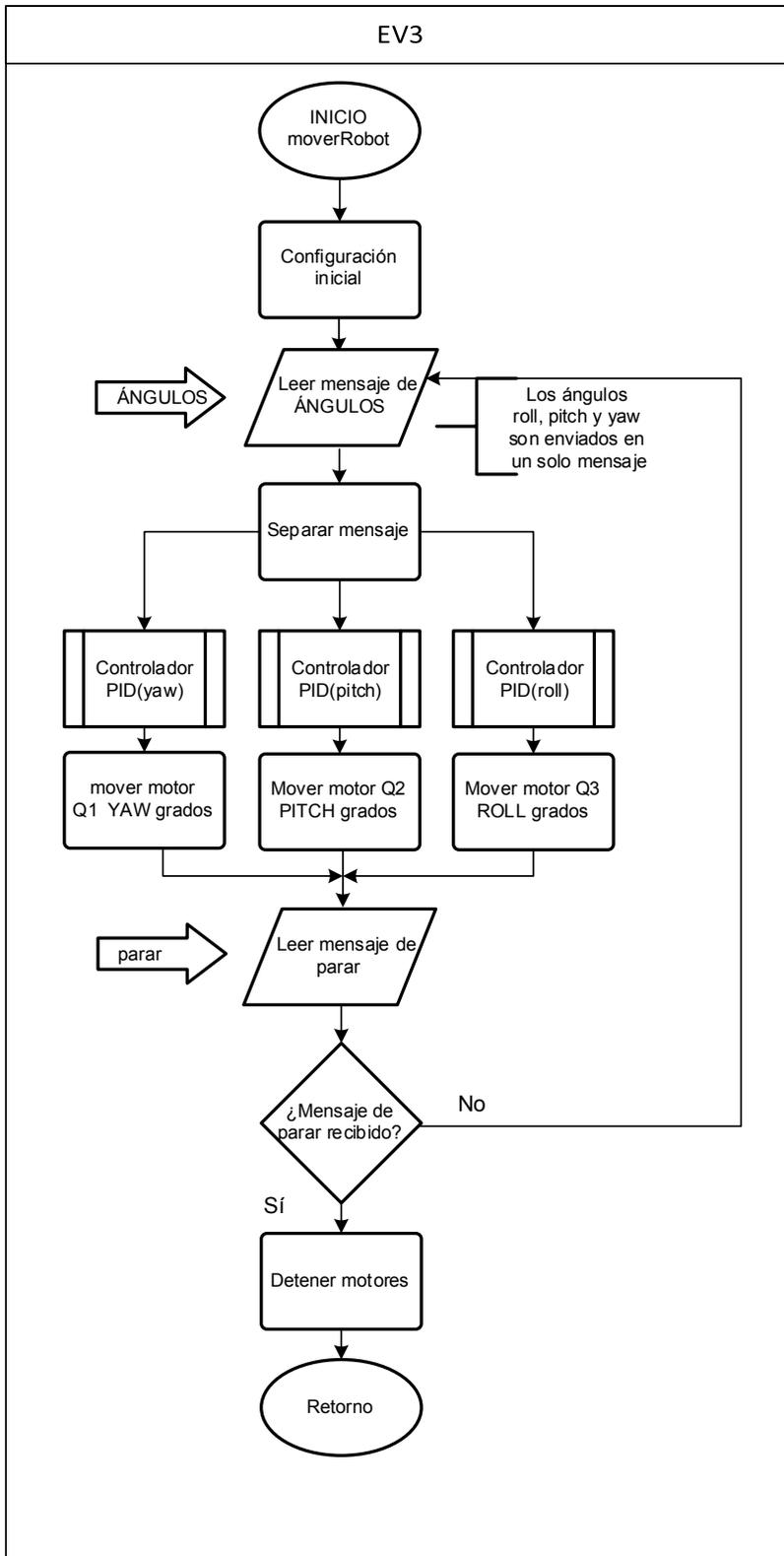


Figura IV.5 Rutina de movimiento del brazo robótico

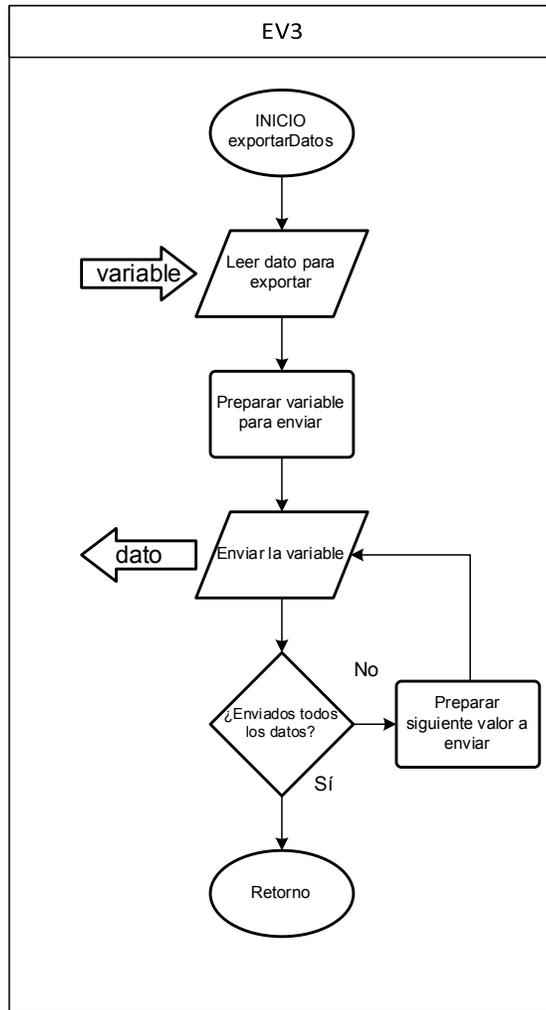


Figura IV.6 Rutina para exportar datos del ladrillo inteligente EV3

ANEXO V. RUTINA DE PRUEBAS

Detalles de los usuarios

En este anexo se incluye los resultados de la rutina de pruebas del sistema de clasificación de gestos. La rutina de pruebas se realizó con diez usuarios que realizaron 30 repeticiones de cada gesto. Entre los diez usuarios hubo 8 hombres y dos mujeres, en el rango de edades entre 22 a 30 años. Todos los usuarios vistieron el Myo Armband en la mano derecha (solo uno de ellos era zurdo). Ninguno de ellos reportó tener algún problema muscular en su brazo, y solo uno de ellos reportó haber sufrido algún tipo de lesión.

Matrices de confusión de cada usuario

Matriz de Confusión. Usuario: alejandro

Predicción del Sistema	Gesto Realizado						
	no Gesto	Wave In	Wave Out	Fist	Open	Pinch	
no Gesto	0 0.0%	4 2.7%	0 0.0%	2 1.3%	1 0.7%	3 2.0%	0.0% 100%
Wave In	0 0.0%	24 16.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
Wave Out	0 0.0%	0 0.0%	30 20.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
Fist	0 0.0%	0 0.0%	0 0.0%	16 10.7%	0 0.0%	0 0.0%	100% 0.0%
Open	0 0.0%	2 1.3%	0 0.0%	12 8.0%	29 19.3%	4 2.7%	61.7% 38.3%
Pinch	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	23 15.3%	100% 0.0%
	NaN% NaN%	80.0% 20.0%	100% 0.0%	53.3% 46.7%	96.7% 3.3%	76.7% 23.3%	81.3% 18.7%

Figura V.1i Resultados usuario: alejandro

Matriz de Confusión. Usuario: alexandra

Predicción del Sistema	no Gesto	0 0.0%	1 0.7%	1 0.7%	4 2.7%	0 0.0%	1 0.7%	0.0% 100%
	Wave In	0 0.0%	26 17.3%	0 0.0%	1 0.7%	0 0.0%	0 0.0%	96.3% 3.7%
	Wave Out	0 0.0%	2 1.3%	29 19.3%	1 0.7%	0 0.0%	0 0.0%	90.6% 9.4%
	Fist	0 0.0%	0 0.0%	0 0.0%	22 14.7%	0 0.0%	0 0.0%	100% 0.0%
	Open	0 0.0%	1 0.7%	0 0.0%	2 1.3%	30 20.0%	0 0.0%	90.9% 9.1%
	Pinch	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	29 19.3%	100% 0.0%
		NaN% NaN%	96.7% 13.3%	96.7% 3.3%	73.3% 26.7%	100% 0.0%	96.7% 3.3%	90.7% 9.3%
		no Gesto	Wave In	Wave Out	Fist	Open	Pinch	
		Gesto Realizado						

Figura V.2 Resultados usuario: Alexandra

Matriz de Confusión. Usuario: andres

Predicción del Sistema	no Gesto	0 0.0%	2 1.3%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0.0% 100%
	Wave In	0 0.0%	28 18.7%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	Wave Out	0 0.0%	0 0.0%	30 20.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	Fist	0 0.0%	0 0.0%	0 0.0%	30 20.0%	0 0.0%	0 0.0%	100% 0.0%
	Open	0 0.0%	0 0.0%	0 0.0%	0 0.0%	30 20.0%	2 1.3%	93.8% 6.3%
	Pinch	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	28 18.7%	100% 0.0%
		NaN% NaN%	93.3% 6.7%	100% 0.0%	100% 0.0%	100% 0.0%	93.3% 6.7%	97.3% 2.7%
		no Gesto	Wave In	Wave Out	Fist	Open	Pinch	
		Gesto Realizado						

Figura V.3 Resultados usuario: andres

Matriz de Confusión. Usuario: andresGuerra

Predicción del Sistema	no Gesto	0 0.0%	1 0.7%	0 0.0%	0 0.0%	6 4.0%	1 0.7%	0.0% 100%
	Wave In	0 0.0%	26 17.3%	0 0.0%	4 2.7%	0 0.0%	1 0.7%	83.9% 16.1%
	Wave Out	0 0.0%	0 0.0%	30 20.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	Fist	0 0.0%	1 0.7%	0 0.0%	23 15.3%	1 0.7%	0 0.0%	92.0% 8.0%
	Open	0 0.0%	0 0.0%	0 0.0%	3 2.0%	23 15.3%	0 0.0%	88.5% 11.5%
	Pinch	0 0.0%	2 1.3%	0 0.0%	0 0.0%	0 0.0%	28 18.7%	93.3% 6.7%
		NaN% NaN%	86.7% 13.3%	100% 0.0%	76.7% 23.3%	76.7% 23.3%	93.3% 6.7%	86.7% 13.3%
		no Gesto	Wave In	Wave Out	Fist	Open	Pinch	
Gesto Realizado								

Figura V.4 Resultados usuario: andresGuerra

Matriz de Confusión. Usuario: cristhian

Predicción del Sistema	no Gesto	0 0.0%	0 0.0%	8 5.3%	0 0.0%	7 4.7%	11 7.3%	0.0% 100%
	Wave In	0 0.0%	30 20.0%	0 0.0%	0 0.0%	0 0.0%	1 0.7%	96.8% 3.2%
	Wave Out	0 0.0%	0 0.0%	19 12.7%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	Fist	0 0.0%	0 0.0%	0 0.0%	30 20.0%	0 0.0%	0 0.0%	100% 0.0%
	Open	0 0.0%	0 0.0%	3 2.0%	0 0.0%	23 15.3%	0 0.0%	88.5% 11.5%
	Pinch	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	18 12.0%	100% 0.0%
		NaN% NaN%	100% 0.0%	63.3% 36.7%	100% 0.0%	76.7% 23.3%	60.0% 40.0%	80.0% 20.0%
	no Gesto	Wave In	Wave Out	Fist	Open	Pinch		
Gesto Realizado								

Figura V.5 Resultados usuario: cristhian

Matriz de Confusión. Usuario: dianita

Predicción del Sistema	no Gesto	0 0.0%	1 0.7%	13 8.7%	3 2.0%	4 2.7%	5 3.3%	0.0% 100%
	Wave In	0 0.0%	27 18.0%	0 0.0%	0 0.0%	0 0.0%	1 0.7%	96.4% 3.6%
	Wave Out	0 0.0%	2 1.3%	15 10.0%	0 0.0%	0 0.0%	0 0.0%	88.2% 11.8%
	Fist	0 0.0%	0 0.0%	0 0.0%	26 17.3%	0 0.0%	0 0.0%	100% 0.0%
	Open	0 0.0%	0 0.0%	2 1.3%	0 0.0%	26 17.3%	0 0.0%	92.9% 7.1%
	Pinch	0 0.0%	0 0.0%	0 0.0%	1 0.7%	0 0.0%	24 16.0%	96.0% 4.0%
		NaN% NaN%	90.0% 10.0%	50.0% 50.0%	86.7% 13.3%	86.7% 13.3%	80.0% 20.0%	78.7% 21.3%
		no Gesto	Wave In	Wave Out	Fist	Open	Pinch	
Gesto Realizado								

Figura V.6 Resultados usuario: dianita

Matriz de Confusión. Usuario: homero

Predicción del Sistema	no Gesto	0 0.0%	0 0.0%	0 0.0%	3 2.0%	13 8.7%	13 8.7%	0.0% 100%
	Wave In	0 0.0%	30 20.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	Wave Out	0 0.0%	0 0.0%	30 20.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	Fist	0 0.0%	0 0.0%	0 0.0%	27 18.0%	0 0.0%	0 0.0%	100% 0.0%
	Open	0 0.0%	0 0.0%	0 0.0%	0 0.0%	8 5.3%	0 0.0%	100% 0.0%
	Pinch	0 0.0%	0 0.0%	0 0.0%	0 0.0%	9 6.0%	17 11.3%	65.4% 34.6%
		NaN% NaN%	100% 0.0%	100% 0.0%	90.0% 10.0%	26.7% 73.3%	56.7% 43.3%	74.7% 25.3%
	no Gesto	Wave In	Wave Out	Fist	Open	Pinch		
Gesto Realizado								

Figura V.7 Resultados usuario: homero

Matriz de Confusión. Usuario: jona

Predicción del Sistema	no Gesto	0 0.0%	0 0.0%	7 4.7%	0 0.0%	0 0.0%	1 0.7%	0.0% 100%
	Wave In	0 0.0%	30 20.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	Wave Out	0 0.0%	0 0.0%	23 15.3%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	Fist	0 0.0%	0 0.0%	0 0.0%	30 20.0%	0 0.0%	0 0.0%	100% 0.0%
	Open	0 0.0%	0 0.0%	0 0.0%	0 0.0%	30 20.0%	0 0.0%	100% 0.0%
	Pinch	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	29 19.3%	100% 0.0%
		NaN% NaN%	100% 0.0%	76.7% 23.3%	100% 0.0%	100% 0.0%	96.7% 3.3%	94.7% 5.3%
	no Gesto	Wave In	Wave Out	Fist	Open	Pinch		
Gesto Realizado								

Figura V.8 Resultados usuario: jona

Matriz de Confusión. Usuario: juan

Predicción del Sistema	no Gesto	0 0.0%	0 0.0%	7 4.7%	1 0.7%	0 0.0%	0 0.0%	0.0% 100%
	Wave In	0 0.0%	30 20.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	Wave Out	0 0.0%	0 0.0%	23 15.3%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
	Fist	0 0.0%	0 0.0%	0 0.0%	29 19.3%	0 0.0%	0 0.0%	100% 0.0%
	Open	0 0.0%	0 0.0%	0 0.0%	0 0.0%	30 20.0%	0 0.0%	100% 0.0%
	Pinch	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	30 20.0%	100% 0.0%
		NaN% NaN%	100% 0.0%	76.7% 23.3%	96.7% 3.3%	100% 0.0%	100% 0.0%	94.7% 5.3%
	no Gesto	Wave In	Wave Out	Fist	Open	Pinch		
Gesto Realizado								

Figura V.9 Resultados usuario: juan

Matriz de Confusión. Usuario: santiago

Predicción del Sistema	Gesto Realizado						
	no Gesto	Wave In	Wave Out	Fist	Open	Pinch	
no Gesto	0 0.0%	0 0.0%	0 0.0%	0 0.0%	1 0.7%	0 0.0%	0.0% 100%
Wave In	0 0.0%	27 18.0%	0 0.0%	0 0.0%	1 0.7%	1 0.7%	93.1% 6.9%
Wave Out	0 0.0%	0 0.0%	26 17.3%	0 0.0%	0 0.0%	0 0.0%	100% 0.0%
Fist	0 0.0%	0 0.0%	0 0.0%	19 12.7%	0 0.0%	0 0.0%	100% 0.0%
Open	0 0.0%	0 0.0%	0 0.0%	0 0.0%	21 14.0%	0 0.0%	100% 0.0%
Pinch	0 0.0%	3 2.0%	4 2.7%	11 7.3%	7 4.7%	29 19.3%	53.7% 46.3%
	NaN NaN	90.0% 10.0%	86.7% 13.3%	63.3% 36.7%	70.0% 30.0%	96.7% 3.3%	81.3% 18.7%

Figura V.10 Resultados usuario: santiago

Respuestas para diferentes entradas paso del primer grado de libertad (yaw)

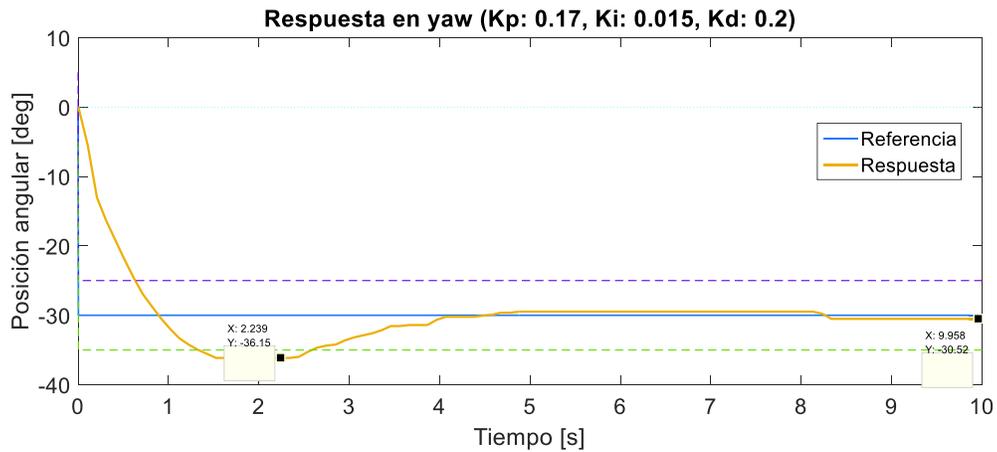


Figura V.11 Respuesta del primer grado de libertad para una entrada paso de -30 grados

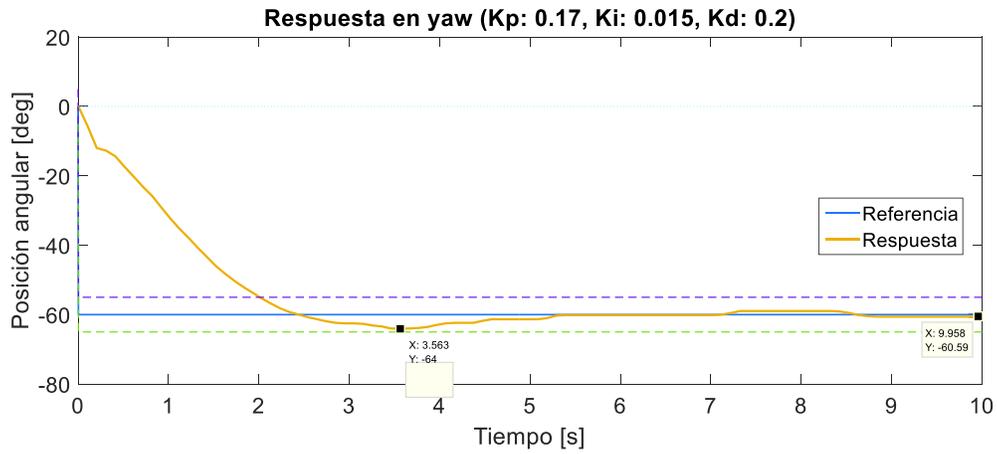


Figura V.12 Respuesta del primer grado de libertad para una entrada paso de -60 grados

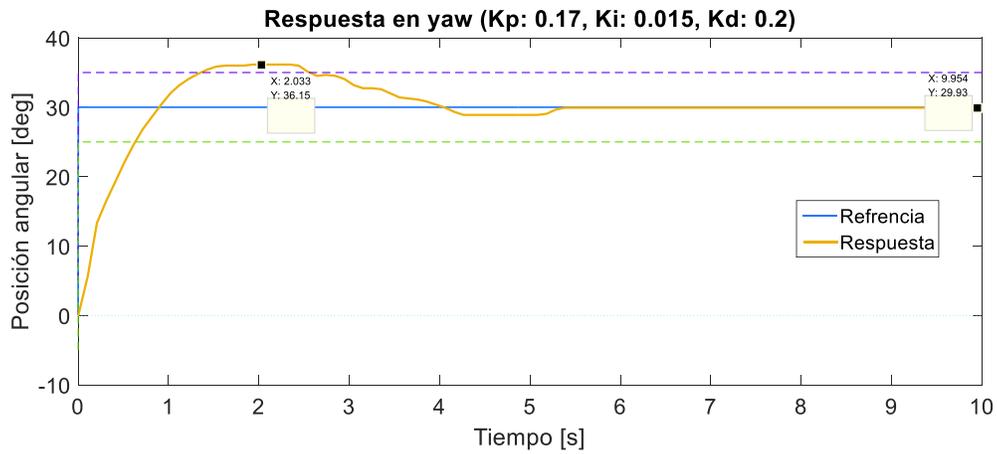


Figura V.13 Respuesta del primer grado de libertad para una entrada paso de +30 grados

Respuestas para diferentes entradas paso del segundo grado de libertad (*pitch*)

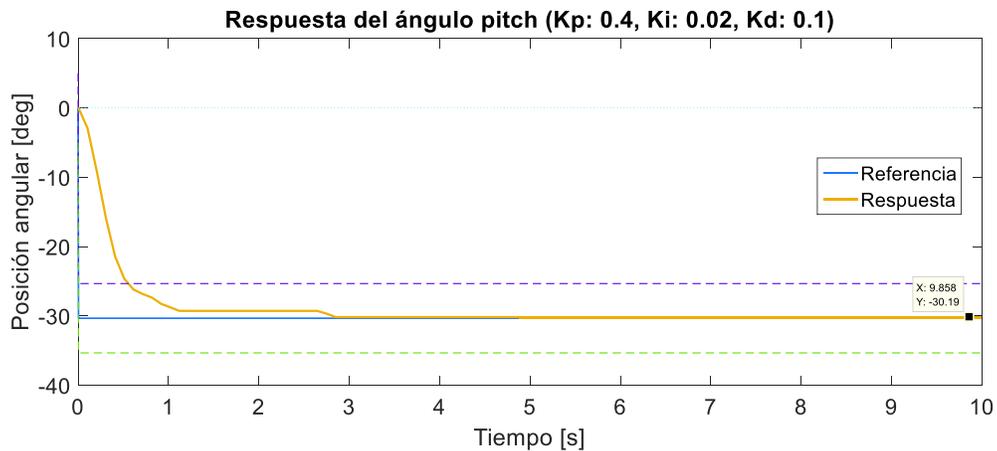


Figura V.14 Respuesta del segundo grado de libertad para una entrada paso de -30 grados

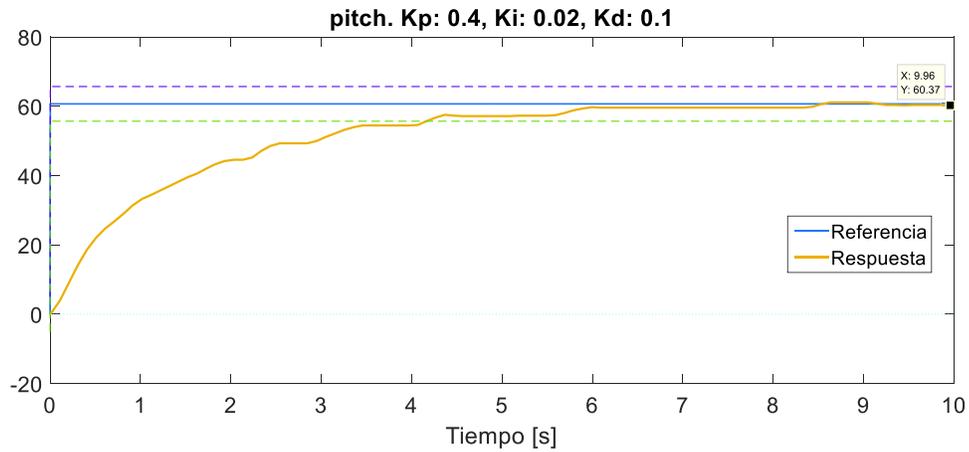


Figura V.15 Respuesta del segundo grado de libertad para una entrada paso de +60 grados

Respuestas para diferentes entradas paso del tercer grado de libertad (roll)

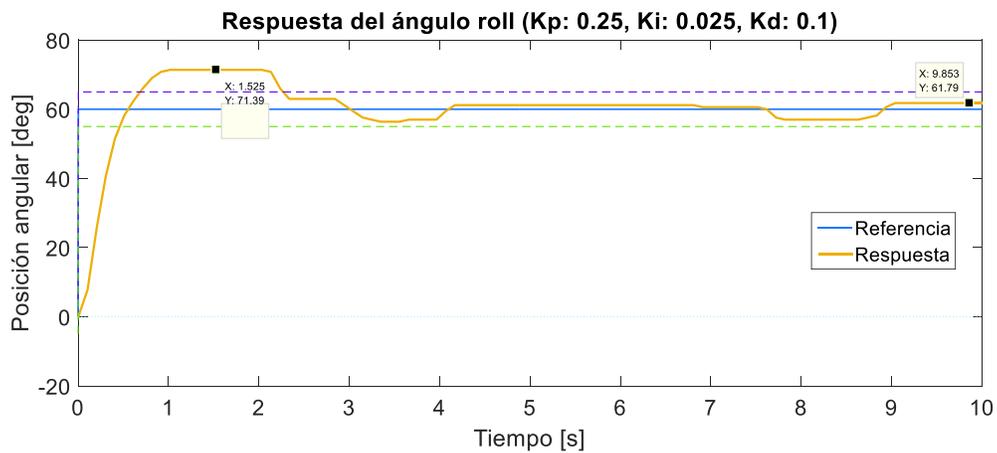


Figura V.16 Respuesta del tercer grado de libertad para una entrada paso de +60 grados

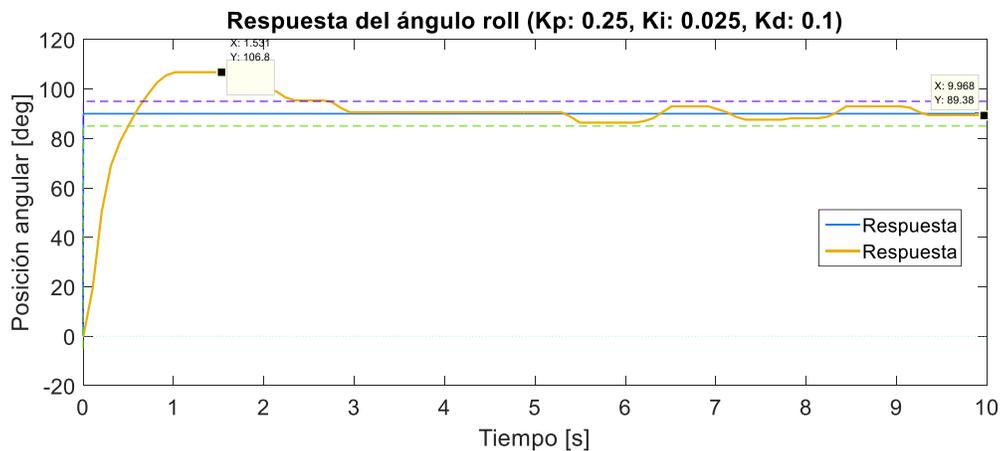


Figura V.17 Respuesta del tercer grado de libertad para una entrada paso de +90 grados

ORDEN DE EMPASTADO