



ESCUELA POLITÉCNICA NACIONAL



FACULTAD DE INGENIERÍA MECÁNICA

OPERACIÓN REMOTA Y CONTROL ADAPTATIVO DE UN MANIPULADOR VIRTUAL DE 3 GRADOS DE LIBERTAD USANDO 2 BRAZALETES MYO Y RECONOCIMIENTO DE GESTOS DE LA MANO

TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE
MAGÍSTER EN MECATRÓNICA Y ROBÓTICA

ING. ALEX FABRICIO OÑA ÑACATA
on_alex8_cp@hotmail.es

ING. VICTOR HUGO VIMOS TIXI
victor.vimos@yahoo.es

DIRECTOR: ING. MARCO ENRIQUE BENALCÁZAR PALACIOS, PhD.
marco.benalcazar@epn.edu.ec

CODIRECTOR: ING. PATRICIO JAVIER CRUZ DÁVALOS, PhD.
patricio.cruz@epn.edu.ec

Quito, Febrero, 2019

CERTIFICACIÓN

Certificamos que el presente trabajo fue desarrollado por **Alex Fabricio Oña Ñacata** y **Victor Hugo Vimos Tixi**, bajo nuestra supervisión.

Ing. Marco E. Benalcázar, PhD
DIRECTOR DEL PROYECTO

Ing. Patricio J. Cruz, PhD.
CODIRECTOR DEL PROYECTO

DECLARACIÓN

Nosotros, **Alex Fabricio Oña Ñacata** y **Victor Hugo Vimos Tixi**, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedemos nuestros derechos de propiedad intelectual correspondiente a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normativa institucional vigente.

Ing. Alex Fabricio Oña Ñacata

Ing. Victor Hugo Vimos Tixi

DEDICATORIA

A Dios, por ser mi fortaleza espiritual, que me ha permitido superar todas las dificultades en la vida.

A mi madre por ser una gran persona luchadora y humilde, que constantemente me brinda su apoyo y consejos que han sido primordial en mi vida. Siendo ella mi gran ejemplo y mi motivación para seguir creciendo como persona y profesionalmente.

A mis queridas hermanas que me han apoyado y dado su amor, que pese a las dificultades siempre me alegran la vida.

A mi padre y hermano que se han preocupado por mí y me han brindado su apoyo.

A mi familia y personas especiales en mi vida, por darme su cariño y apoyo incondicional.

A mi primo Ronny en el cielo, que siempre permanece en mi corazón.

Alex

A toda mi familia por su apoyo constante e incondicional. A mi papi Víctor por sus grandes consejos, a mi mami Teresa por transmitirme su perseverancia cada día, a mí hermana Alexandra por su ejemplo de valentía, a mi abuelito Juan por sus grandes lecciones de vida, a mí novia Gaby por su cariño y amor en todo momento.

A mis queridos familiares que ya no están, que siempre fueron un motivo diario de superación y fuente inagotable de buenos y sabios consejos.

Victor

AGRADECIMIENTOS

Agradezco a Dios por haberme dado a mi familia y a personas especiales en mi vida. Por permitirme aprender nuevas cosas y que a pesar de pasar momentos adversos siempre pone en mi corazón este hermoso mensaje: *“Mira que te mando que te esfuerces y seas valiente; no temas ni desmayes. Porque Jehová tu Dios estará contigo a dondequiera que vayas”* (Josué 1:9).

Agradezco a mi madre por darme la vida, por su gran amor y su constante lucha para formarme como un hombre de bien. A mi padre por haberme dado la vida, y brindarme su ayuda siempre.

A mis hermanas Mayri y Taty por ser unas mujeres especiales en mi vida, por su apoyo incondicional y ser mi motivación permanente. Así mismo agradezco la motivación y preocupación de mi hermano Jorge en cada uno de mis proyectos emprendidos.

A mi amor Johita por su amor, apoyo y por los momentos especiales que hemos compartido juntos. A toda mi familia, mis queridos abuelitos, tíos, tías, primos, primas y amigos cercanos que siempre están pendientes de mi persona.

Agradezco a los doctores Patricio Cruz y Marco Benalcázar por su respaldo en el desarrollo del presente proyecto, al igual que por brindarme su valioso tiempo y paciencia para lograr hacer un buen trabajo.

A mi amigo Víctor Vimos por su amistad a lo largo de este Postgrado y en la realización del presente trabajo.

Alex

A toda mi familia, amigos y maestros con los que he aprendido maravillosas lecciones en las aulas y fuera de ellas.

Un especial agradecimiento al Doctor Marco Benalcázar y al Doctor Patricio Cruz por su apoyo constante para poder realizar un buen trabajo de tesis. De igual manera un agradecimiento especial a mi buen amigo Alex Oña por su ayuda en la realización de este trabajo de tesis.

Victor

ÍNDICE

INTRODUCCIÓN	15
Objetivo general	16
Objetivos específicos	16
Alcance	17
1. MARCO TEÓRICO	19
1.1. Robótica.....	19
1.1.1. Robot	19
1.1.2. Clasificación	19
1.2. Manipulador robótico	19
1.2.1. Configuraciones cinemáticas	21
1.2.2. Volumen de trabajo	21
1.2.3. Sistemas funcionales de un manipulador	21
1.3. Robots operados remotamente.....	22
1.4. Plataforma robótica seleccionada.....	24
1.5. Modelamiento CAD 3D de la plataforma robótica	26
1.5.1. Elementos modelados.....	26
1.5.2. Ensamblés.....	27
1.5.3. Espacio de trabajo.....	28
1.5.4. Parámetros del manipulador	29
1.6. Cinemática de manipuladores.....	30
1.6.1. Problema de la cinemática directa	30
1.6.2. Problema de la cinemática inversa	33
1.6.3. Jacobiano	33
1.7. Dinámica del manipulador.....	35
1.7.1. Obtención de las ecuaciones dinámicas del manipulador	36
1.8. Control de manipuladores	40
1.8.1. Control de posición.....	40
1.8.2. Control de trayectoria	40
1.8.3. Control por par calculado – CPC	42
1.8.4. Control de inercia adaptativa – CIA	42
1.9. Miología	43
1.9.1. Músculos del antebrazo	43
1.9.2. Electromiografía	45
1.10. MYO Armband.....	46

1.11. Reconocimiento de gestos	48
1.12. Etapas de los sistemas de clasificación de gestos.....	48
1.12.1. Adquisición	49
1.12.2. Pre-procesado y extracción de características.....	50
1.12.3. Clasificación	51
2. METODOLOGÍA.....	53
2.1. Adquisición de datos con el brazalete MYO Armband	53
2.2. Selección de datos de sensores IMU	54
2.2.1. Asignación de ángulos para control de movimiento	55
2.3. Cinemática inversa	56
2.4. Diseño de controladores	58
2.4.1. Controlador por par calculado (CPC).....	58
2.4.2. Controlador de inercia adaptativa (CIA)	60
2.5. Gestos para control del manipulador virtual	65
2.5.1. Wave Out.....	65
2.5.2. Wave In.....	66
2.5.3. Close.....	67
2.5.4. Thumb.....	68
2.5.5. Relax.....	69
2.5.6. Index.....	70
2.6. Medidas estadísticas y extracción de características	71
2.6.1. Valor medio	71
2.6.2. Desviación estándar	72
2.6.3. Envoltente de la curva EMG.....	72
2.6.4. Cruces por cero	73
2.6.5. Energía	73
2.6.6. Densidad espectral de potencia.....	73
2.6.7. Valor medio absoluto.....	73
2.7. Algoritmos de clasificación	74
2.7.1. Máquina de vectores de soporte (SVM)	74
2.7.2. Vecinos más próximos (KNN).....	75
2.7.3. Árbol de decisiones	75
2.8. Matrices de datos de entrenamiento y testeo.....	75
2.9. Matlab Classification Learner.....	78
2.9.1. Entrenamiento	80
2.10. HMI de control del brazo robótico	83

2.10.1. Interfaz principal (MYO 1)	83
2.10.2. Interfaz MYO 2	88
2.11. Implementación del entorno virtual	88
2.12. Sistema de operación remota	89
3. PRUEBAS Y RESULTADOS	92
3.1. Pruebas off-line del sistema robótico virtual (Esclavo).....	92
3.1.1. Prueba de simulación con referencia tipo escalón para CPC y CIA.....	93
3.1.2. Prueba de simulación con referencia sinusoidal para CPC y CIA.....	95
3.1.3. Prueba de simulación con referencia sinusoidal y variación de masa.....	97
3.2. Pruebas off-line de la interfaz de control (maestro).....	104
3.3. Pruebas on-line para operación remota del robot virtual	106
3.3.1. Consideraciones.....	106
3.3.2. Metodología.....	108
3.3.3. Resultados.....	109
4. CONCLUSIONES Y RECOMENDACIONES	114
4.1. Conclusiones	114
4.2. Recomendaciones.....	115
Referencias Bibliográficas	117
Anexos	121

ÍNDICE DE FIGURAS

Figura 1.1. Cadena cinemática: (a) abierta y (b) cerrada.....	20
Figura 1.2. Configuración cinemática angular para un robot de 3GDL.....	21
Figura 1.3. a) Base fija; b) Tapa giratoria; c) Rodamiento.	26
Figura 1.4. Soportes multipropósito para servomotor: a) ASB204; b) ASB201.....	26
Figura 1.5. Soportes largo tipo C para servomotor: a) ASB203; b) ASB205 y c) Paneles tipo disco.....	26
Figura 1.6. a) Soporte tipo L; b) Conector; c) Tubo central.....	27
Figura 1.7. Servomotores: a) HS-485HB, b) HS-805BB, c) HS-755.....	27
Figura 1.8. a) Base fija; b) Hombro.....	27
Figura 1.9. a) Brazo; b) Antebrazo.....	27
Figura 1.10. Manipulador robótico 3GDL.....	28
Figura 1.11. Vistas: a) lateral izquierda y b) superior, presentadas con dimensiones en milímetros.	28
Figura 1.12. Asignación de sistemas de referencia.....	30
Figura 1.13. Unidad motora.....	43
Figura 1.14. Flexor común profundo de los dedos.	44
Figura 1.15. Músculo pronador redondo.....	44
Figura 1.16. Músculo cubital anterior.....	45
Figura 1.17. Músculo extensor radial corto.....	45
Figura 1.18. Registro de EMG de acción de una unidad motora.	46
Figura 1.19. a) Brazaletes MYO; b) Brazaletes MYO en el antebrazo de una persona.	47
Figura 1.20. Gestos reconocidos por default en el brazaletes MYO.	47
Figura 1.21. Etapas del sistema de clasificación de gestos.....	49
Figura 1.22. Configuraciones de electrodos para EMG: a) mono-polar y b) bi-polar.	49
Figura 1.23. Configuración bi-polar de electrodos del brazaletes MYO.	49
Figura 1.24. a) EMG sin filtrar, b) EMG filtrada $f_c=50\text{Hz}$	50
Figura 1.25. a) Sistema separable linealmente, b) Sistema no separable linealmente.....	51
Figura 2.1. Software Myo Connect durante la instalación.....	53
Figura 2.2. Gestos predefinidos que reconoce el brazaletes MYO.	53
Figura 2.3. a) Postura inicial; b) ubicación del MYO en el brazo; c) ubicación del MYO en el antebrazo.....	55
Figura 2.4. Asignación de ángulos: a) Y_1 ; b) P_1	55
Figura 2.5. Asignación del ángulo P_2	56
Figura 2.6. Plano generado por P_1 y P_2	56

Figura 2.7. Triángulo con Pitch 1; Triángulo con Pitch 2.....	57
Figura 2.8. Plano 2D del robot.	57
Figura 2.9. Diagrama de bloques del controlador CPC.....	58
Figura 2.10. Diagrama de bloques del controlador CIA.	63
Figura 2.11. Gesto wave out, se le asocia el número “1”.....	65
Figura 2.12. EMG del gesto wave out, señal filtrada a una frecuencia de corte de 50Hz. 66	
Figura 2.13. Gesto wave in, se le asocia el número “2”.	66
Figura 2.14. EMG del gesto wave in, señal filtrada a una frecuencia de corte de 50Hz. ..	67
Figura 2.15. Gesto close, se le asocia el número “3”.	67
Figura 2.16. EMG del gesto close, señal filtrada a una frecuencia de corte de 50Hz.	68
Figura 2.17. Gesto thumb, se le asocia el número “5”.	68
Figura 2.18. EMG del gesto thumb, señal filtrada a una frecuencia de corte de 50Hz.	69
Figura 2.19. Gesto relax, se le asocia el número “6”.....	69
Figura 2.20. EMG del gesto relax, señal filtrada a una frecuencia de corte de 50Hz.	70
Figura 2.21. Gesto index, se le asocia el número “7”.	70
Figura 2.22. EMG del gesto index, señal filtrada a una frecuencia de corte de 50Hz.....	71
Figura 2.23. Envoltente de la curva EMG (color naranja).	72
Figura 2.24. EMG organizadas por sensor.	74
Figura 2.25. Sensor de referencia para usuario 2.	77
Figura 2.26. Opción para pruebas, Holdout Validation.....	79
Figura 2.27. Características ingresadas para la clasificación de gesto.	79
Figura 2.28. Matriz de confusión con una exactitud del 100%, clasificador SVM usando la base de datos de entrenamiento.....	80
Figura 2.29. Respuesta obtenida vs cantidad de repeticiones de gestos, 6 usuarios, SVM clasificador.....	80
Figura 2.30. Porcentaje según el gesto usando SVM para los 6 usuarios.	80
Figura 2.31. Matriz de confusión con una exactitud del 98.6%, clasificador KNN usando la base de datos de entrenamiento.....	81
Figura 2.32. Respuesta obtenida vs cantidad de repeticiones de gestos, 6 usuarios, KNN clasificador.....	81
Figura 2.33. Porcentaje según el gesto usando KNN para los 6 usuarios.	81
Figura 2.34. Matriz de confusión con una exactitud del 86.1%, clasificador árbol de decisiones usando la base de datos de entrenamiento.	82
Figura 2.35. Respuesta obtenida vs cantidad de repeticiones de gestos, 6 usuarios, clasificador árbol de decisión.	82
Figura 2.36. Porcentaje según el gesto usando árbol de decisión para los 6 usuarios. ..	82

Figura 2.37. Interfaz MYO 1, pestaña de calibración. a) Instalar Myo, b) Indicador de sensor de referencia, c) Inicio de adquisición, d) Graficación EMG, e) Seteo sensor de referencia.	83
Figura 2.38. Distribución de los sensores en el brazalete MYO.	83
Figura 2.39. Interfaz MYO 1, pestaña de entrenamiento. a) Selección del gesto a entrenar, b) Graficación EMG, c) Guardar datos adquiridos, d) Gesto actual de entrenamiento.	84
Figura 2.40. Interfaz MYO 1, pestaña de testeo. a) Gesto detectado, b) Habilitar el reconocimiento de gestos.	85
Figura 2.41. Interfaz MYO 1, pestaña de selección de herramientas y colores. a) Indicador de la característica actual, b) Muestra el peso y color seleccionado.	85
Figura 2.42. Interfaz MYO 1, pestaña de conexión remota y control del brazo robótico. a) Posición actual del brazo, b) Seteo del punto de referencia de los brazaletes, c) Tipo de conexión con el brazo virtual, d) Adquisición de datos IMU del brazalete 2.	87
Figura 2.43. Interfaz de conexión del brazalete MYO 2.	88
Figura 2.44. Robot manipulador en el entorno virtual.	89
Figura 2.45. Sistema para operación remota.	90
Figura 2.46. Realimentación visual vMix.	91
Figura 2.47. Sistema de realimentación con cámara IP (Celular).	91
Figura 3.1. a) Respuesta para diagonal de Γ con 5; b) Respuesta para diagonal de Γ con 100.	92
Figura 3.2. Seguimiento y error de posición angular de la articulación del hombro (q_1) - CPC y CIA.	93
Figura 3.3. Seguimiento y error de posición angular de la articulación del brazo (q_2) - CPC y CIA.	93
Figura 3.4. Seguimiento y error de posición angular de la articulación del antebrazo (q_3) - CPC y CIA.	94
Figura 3.5. Torque aplicado a las articulaciones del hombro y brazo (q_1, q_2) – CPC y CIA.	94
Figura 3.6. Torque aplicado a la articulación del antebrazo (q_3) – CPC y CIA.	94
Figura 3.7. Posición angular de la articulación del brazo (q_2) para CPC y CIA.	95
Figura 3.8. Errores de posición y velocidad de la articulación del brazo (q_2) para CPC y CIA.	96
Figura 3.9. Torque aplicado a la articulación del brazo (q_2) para CPC y CIA.	96
Figura 3.10. Trayectoria efectuada por el extremo del robot en el plano XY y en el espacio 3D.	96

Figura 3.11. Evolución temporal de la articulación del brazo (q_2) para CPC y CIA (masa de objeto 5g).....	97
Figura 3.12. Errores de posición y velocidad de la articulación del brazo (q_2) para CPC y CIA (masa de objeto 5g).	98
Figura 3.13. Torque aplicado a la articulación del brazo (q_2) para CPC y CIA (masa de objeto 5g).....	98
Figura 3.14. Estimación de masas de los eslabones en el controlador CIA por medio de la Ecuación 2.42 (masa de objeto 5g).	98
Figura 3.15. Trayectoria efectuada por el extremo del robot en el plano XY y en el espacio 3D (masa de objeto 5g).....	99
Figura 3.16. Evolución temporal de la articulación del brazo (q_2) para CPC y CIA (masa objeto 10g).....	99
Figura 3.17. Errores de posición y velocidad de la articulación del brazo (q_2) para CPC y CIA (masa de objeto 10g).	100
Figura 3.18. Torque aplicado a la articulación del brazo (q_2) para CPC y CIA (masa de objeto 10g).....	100
Figura 3.19. Estimación de masas de los eslabones en el controlador CIA por medio de la Ecuación 2.42 (masa de objeto 10g).	100
Figura 3.20. Trayectoria efectuada por el extremo del robot en el plano XY y el espacio 3D (masa objeto 10g).....	101
Figura 3.21. Evolución temporal de la articulación del brazo (q_2) para CPC y CIA (masa objeto 100-5g).....	101
Figura 3.22. Errores de posición y velocidad de la articulación del brazo (q_2) para CPC y CIA (masa de objeto 100-5g).	102
Figura 3.23. Torque aplicado a la articulación del brazo (q_2) para CPC y CIA (masa de objeto 100-5g).....	102
Figura 3.24. Estimación de masas de los eslabones en el controlador CIA por medio de la Ecuación 2.42 (masa de objeto 100-5g).	102
Figura 3.25. Trayectoria efectuada por el extremo del robot en el plano XY y espacio 3D (masa de objeto 100-5g).	103
Figura 3.26. Prueba del sistema de reconocimiento de gestos.	104
Figura 3.27. Valoración 20 usuarios, gestos Out, Close, Relax, In, Thumb, Index.	106
Figura 3.28. Puntos establecidos en el entorno virtual.....	107
Figura 3.29. Prueba de operación remota.	108

ÍNDICE DE TABLAS

Tabla 1.1. Manipuladores robóticos.....	24
Tabla 1.2. Criterios y ponderaciones para selección.....	25
Tabla 1.3. Matriz para selección del Manipulador.	25
Tabla 1.4. Limitaciones angulares.	29
Tabla 1.5. Parámetros del manipulador virtual.	29
Tabla 1.6. Parámetros DH.....	31
Tabla 1.7. Controladores para manipuladores.	41
Tabla 2.1. Torques máximos de servomotores.	64
Tabla 3.1. Desempeño de controladores mediante IAE e ISE.....	95
Tabla 3.2. Desempeño de controladores mediante IAE e ISE.....	97
Tabla 3.3. Desempeño de controladores mediante IAE e ISE.....	99
Tabla 3.4. Desempeño de controladores mediante IAE e ISE.....	101
Tabla 3.5. Desempeño de controladores mediante IAE e ISE.....	103
Tabla 3.6. Resultados de reconocimiento de 20 usuarios.	105
Tabla 3.7. Valoración del reconocimiento de gestos según los usuarios	105
Tabla 3.8. Tiempo de demora en trasladar el objeto entre los puntos A y B.....	109

RESUMEN

En el presente trabajo se desarrolla un sistema para operar a distancia un manipulador virtual de 3 grados de libertad en un entorno computacional simulado en Matlab. Como interfaz de control se emplea 2 brazaletes MYO del fabricante Thalmic Labs. El brazalete MYO está constituido por sensores electromiográficos (EMG) que registran la actividad de los músculos esqueléticos y una unidad de medida inercial (IMU) que brinda datos de orientación, mismos que han sido utilizados para el control del movimiento del robot. Adicionalmente, mediante el reconocimiento de gestos de la mano se ha podido variar parámetros como masa y color de un objeto sujetado por el manipulador (por ejemplo, realizar un gesto con la mano permite cambiar de herramienta). La simulación del comportamiento del robot se basa en su modelo cinemático y dinámico obtenidos a través de las metodologías establecidas en la bibliografía. Respecto al control de trayectoria, se ha considerado dos controladores: uno basado en el modelo CPC (Control por Par Calculado) que busca eliminar las no linealidades del sistema cuando los parámetros se conocen con exactitud. Otro tipo de control usado es el CIA (Control de Inercia Adaptativa) que brinda mayor robustez adaptándose a los cambios en la masa del extremo del robot que ocurre cuando el manipulador sujeta el objeto, o cuando existe incertidumbre en los parámetros de éste. Para el reconocimiento de gestos se ha realizado un procesamiento de las señales EMG y usando algoritmos de machine learning se ha logrado clasificar 6 patrones que son usados en el control del brazo robótico virtual. Finalmente, las pruebas de operación remota han sido realizadas usando una red local inalámbrica permitiendo validar el reconocimiento de gestos y el control de la plataforma robótica virtual con una realimentación de tipo visual. El sistema completo es probado con 20 personas de diferentes edades, hombres y mujeres que no pertenecen al banco de datos de entrenamiento inicial del sistema. Los resultados obtenidos muestran una exactitud del 91.6% en el sistema de reconocimiento de gesto con un tiempo de clasificación menor a los 30ms. El resultado general del sistema, tomando en cuenta la valoración de los 20 usuarios, muestra que el sistema total en términos de usabilidad tiene un porcentaje de 82.91% de aceptación; mientras que en términos de funcionalidad el sistema total tiene un promedio de 64.79%, este porcentaje es debido a la falta de practica de los usuarios con el sistema.

Palabras clave: control de par calculado, control de inercia adaptativa, EMG, dinámica y cinemática de manipuladores, KNN, operación remota reconocimiento de gestos, SVM.

ABSTRACT

A system to drive a virtual manipulator with 3 degrees of freedom in a simulated computational environment is developed in this work. The control interface uses 2 MYO bracelets from the Thalmic Labs manufacturer. The MYO bracelet gives the activity of the skeletal muscles (EMG) and data from the inertial measurement unit (IMU). Support Vector Machines (SVM) algorithms working with EMG data from MYO bracelet have made possible to vary parameters such as mass and color of an object held by the manipulator. Regarding trajectory control, two controllers have been considered: the CTC model (Calculated Torque Control) that seeks to eliminate the nonlinearities of the system when the parameters are well known. The other control used is AIC (Adaptive Inertia Control) that provides greater robustness adapting to changes in the mass of the final effector robot that occurs when the manipulator holds different objects. EMG signals have been processed and it has been possible to classify 6 gestures which are used to control the virtual robotic arm. Using a wireless local network the complete system is tested allowing validation of gesture recognition and the control of the virtual robotic arm with visual feedback. The system is tested with 20 people who do not belong to the database of training people. The results obtained show an accuracy of 91.6% in the gesture recognition system with a time classification less than 30ms. The overall result taking into account the assessment of the 20 users, shows that the total system in terms of usability has an acceptance of 82.91%, while in functionality the system has an acceptance of 64.79%, this percentage is due to the lack practice with the system.

Keywords: calculated torque control, adaptive inertia control, EMG, dynamic and manipulator kinematics, KNN, remote operation, gesture recognition, SVM.

INTRODUCCIÓN

Dentro de la robótica el campo de manipuladores es el área de mayor estudio y desarrollo por sus grandes prestaciones [1, 2]. Además, presenta grandes retos ya que los manipuladores robóticos son sistemas dinámicos de múltiples entradas, múltiples salidas (MIMO) y de naturaleza no lineal [3, 4]. Esto ha dado origen a tres grandes áreas de interés y de continuo desarrollo: diseño, modelamiento y control de robots [6, 7]. Para éste tipo de plataformas existe varias técnicas de control, desde técnicas de control clásico (por ejemplo, control PID) hasta robustas (por ejemplo, el de inercia adaptativa). Es así que en aplicaciones donde se exige una mayor precisión y robustez de los controladores ante perturbaciones, el control clásico no presenta un buen desempeño, abriendo un campo de estudio y desarrollo de controladores no lineales. El problema de control se complica aún más si hay incertidumbre o variación de los parámetros del robot, por lo cual se requiere de la implementación de controladores más robustos que se adapten a dichas perturbaciones [4,11].

En la actualidad el uso de manipuladores robóticos se ha ido extendiendo y para ciertas aplicaciones es imprescindible la intervención de un operador humano, sobre todo en entornos no estructurados, en donde la planificación automática de las acciones del robot puede resultar compleja. Es ahí donde surge el desarrollo de sistemas operados remotamente (tele-operados) [4]. Estos sistemas tienen varias aplicaciones, por ejemplo: intervención en lugares peligrosos para el ser humano, cirugía robótica (medicina), prótesis inteligentes, industria, entre otros [4, 9]. En estos sistemas, la interfaz de control es un tema de constante desarrollo, pues se ha implementado y probado sistemas utilizando: visión artificial, exoesqueletos, sensores de biopotenciales, sensores inerciales (IMU), entre otros. La utilización de exoesqueletos y de dispositivos basados en visión artificial presentan ciertas dificultades como: el acople mecánico a usuarios o la necesidad de un ambiente lumínico controlado, respectivamente [4, 5, 10, 17]. Esto ha abierto campo a la investigación relacionada con el uso de señales biológicas en tareas de control de dispositivos robóticos.

Una de las características de los sistemas biomecánicos es utilizar señales biológicas para el control de dispositivos mecatrónicos. Un tipo de señal biológica usada son las electromiográficas (EMG), que son señales eléctricas que muestran el comportamiento del sistema biológico muscular que compone el cuerpo humano. Hoy en día, dichas señales son utilizadas en varias áreas de la industria, por ejemplo: control de prótesis, control de equipos mecánicos y electrónicos de alta complejidad y sistemas de tele operación [41,

42]. Esto ha abierto un campo de investigación bastante amplio; es así, como diferentes trabajos se ha desarrollado en esta área [43, 44] y enfocan sus investigaciones en el control de plataformas robóticas, usando técnicas avanzadas para el reconocimiento de patrones usando EMG [45].

En nuestro país se ha incursionado levemente en el campo de robots manipuladores [46], ya que en trabajos como en [17] se usan plataformas comerciales. Además, se ha utilizado generalmente técnicas de control PID sin considerar las no linealidades, los acoplamientos entre los eslabones, e incluso las fuerzas gravitacionales que son evidentes en manipuladores, las cuales pueden ser un problema en ciertas aplicaciones como cuando el manipulador sujeta un objeto cuya masa puede alterar su dinámica [10].

En este proyecto se desarrolla un sistema que permite manejar remotamente un manipulador virtual mediante señales inerciales y de EMG. El manipulador se usa para desplazar una masa de un punto a otro, dicha masa es utilizada como una perturbación a la dinámica del manipulador. Con el fin de presentar robustez ante perturbaciones se implementa un controlador no lineal adaptativo (robusto) cuyo desempeño en base al error de posición de la señal de referencia se compara con un controlador no lineal pero no adaptativo. Para el control del movimiento del manipulador de manera remota, se emplea una interfaz de sensores (IMU) y de EMG embebidos en brazaletes comerciales marca MYO. Se realiza el tratamiento de los datos de IMU's, así como la clasificación y reconocimiento de 6 gestos de la mano usando EMG detectadas en el antebrazo de una persona. Finalmente, por medio de una interfaz gráfica se puede entrenar, calibrar y probar los algoritmos de reconocimientos de gestos, así como realizar la operación remota del manipulador.

Objetivo general

Desarrollar un Sistema para la Operación Remota y Control Adaptativo de un Manipulador Virtual de 3 Grados de Libertad usando Brazaletes MYO y Reconocimiento de Gestos de la Mano.

Objetivos específicos

- ✓ Recopilar información referente a: manipuladores comerciales de 3GDL, dinámica y cinemática de manipuladores, operación remota, procesamiento de señales EMG, clasificadores, controladores no lineales y adaptativos, con el fin de estudiar el estado del arte y poder establecer las técnicas a usar dentro del proyecto.

- ✓ Diseñar e implementar un HMI, algoritmos: para la adquisición y procesamiento de señales EMG y señales IMU de dos brazaletes MYO, que permitan el control de movimiento del manipulador y el reconocimiento de al menos 5 gestos de la mano.
- ✓ Implementar en un sistema computacional un manipulador virtual, y controladores: no lineal no adaptativo y no lineal adaptativo a fin de mejorar el seguimiento de trayectoria ante perturbaciones respecto al no adaptativo.
- ✓ Implementar el sistema de operación remota para el control a distancia del manipulador virtual empleando brazaletes y una red de datos de área local inalámbrica.
- ✓ Evaluar el componente del sistema de control en base al error de posición con respecto a señales de referencia angular; el componente de reconocimiento de gestos a través del tiempo de respuesta menor a 300ms y el porcentaje de acierto de los patrones realizados mayor al 70%, y el componente de operación remota mediante el uso del sistema por un usuario para cumplir una misión en el entorno virtual (off-line, on-line).

Alcance

Para el cumplimiento de los objetivos se ha considerado los siguientes productos:

Contar con un manipulador robótico con 3 GDL modelado en un software CAD (Inventor Autodesk). Dicha estructura al ser usada en el software Matlab (Simulink) que, junto con ecuaciones dinámicas y cinemáticas, permite simular movimientos del robot.

Dentro del análisis de la dinámica del manipulador, no se ha considerado el efecto de fricciones en las articulaciones, pues al no tratarse de un manipulador industrial, dicho efecto puede considerarse despreciable.

Los controladores no lineal y robusto implementados en Simulink, y la prueba del desempeño de éstos se ha realizado por separado. Se tiene 3 valores de referencia que corresponden a las posiciones angulares de las articulaciones del robot, las cuales cambian en función de los comandos enviados por la interfaz de control a través del canal de comunicación en una red de datos WLAN.

En el proyecto se logra controlar los movimientos del manipulador virtual usando información IMU de los 2 brazaletes comerciales MYO, además, se emplea las señales EMG de un brazalete ubicado en el antebrazo para obtener la actividad bioeléctrica de la mano.

Para el censado se usa brazaletes comerciales MYO, estos brazaletes permiten registrar datos de EMG e inerciales los cuales son enviados inalámbricamente hacia el computador. Además, se realizó un procesamiento de las señales y mediante algoritmos de procesamiento y clasificación se efectuó el reconocimiento de 6 gestos de la mano, los cuales son usados para el manejo de un HMI y para poder interactuar remotamente con el robot virtual.

Para la operación remota del manipulador, los movimientos realizados por el usuario establecen referencias de posición en el espacio de trabajo del mismo, dichos movimientos se realizan de manera que el robot virtual realice un seguimiento suave a dichas referencias. No se ha considerado como variable la velocidad de ejecución de los movimientos, debido a los retardos propios de la simulación y, además, esta variable se encuentra limitada por el tiempo de respuesta de sensores y actuadores. El traslado de las masas se realiza entre 2 puntos fijos dentro de su espacio de trabajo, y las masas son representadas por formas esféricas.

1. MARCO TEÓRICO

1.1. Robótica

La robótica es un campo de estudio multidisciplinar y de constante desarrollo, la cual está enfocada en el diseño, análisis y control de robots, en especial de manipuladores, los cuales son diseñados para realizar varias aplicaciones como: industriales, científicas y domésticas. Involucra varias áreas del conocimiento entre ellas: Física, Matemáticas, Electrónica, Inteligencia Artificial, Teoría de Control y muchas más [1, 2, 7].

1.1.1. Robot

La palabra robot tiene muchas definiciones, de manera que, en este trabajo se ha optado por el concepto tomado del Instituto Norteamericano del Robot y de la ISO 8373, que menciona:

“Un robot es un manipulador reprogramable, multifuncional, controlado automáticamente, que puede estar fijo en un sitio o moverse, y que está diseñado para mover objetos mediante movimientos variables programados para la realización de diversas tareas o trabajos” [2].

Frecuentemente se suele añadir un adjetivo al término robot con el objetivo de dar a entender de mejor manera sus características o aplicaciones, por ejemplo, robot: manipulador, humanoide, doméstico, aéreo, tele-operado, bio-médicos, entre otros [2].

1.1.2. Clasificación

Actualmente, existe una variedad de robots, los cuales son clasificados en función de diferentes criterios o características, por ejemplo: generación, área de aplicación, tipo de actuadores, número de ejes, configuración, tipo de control, entre otros. Siendo de interés para este trabajo el manipulador robótico de configuración cinemática angular.

1.2. Manipulador robótico

La robótica industrial puede definirse como el estudio, diseño y uso de robots para la ejecución de procesos industriales. Generalmente los robots industriales tienen semejanza a un brazo humano, es por eso que son considerados como brazos mecánicos articulados o manipuladores robóticos capaces de desempeñar actividades específicas. De manera general un manipulador consiste en un brazo compuesto por elementos conectados por medio de articulaciones, en el cual en el último enlace se coloca un elemento denominado efector final, el mismo que puede ser una herramienta que permite cumplir con una actividad [1, 2].

- **Eslabón:** también denominado como “link”, son elementos que conforman la estructura del robot, y generalmente son considerados como cuerpos rígidos.
- **Articulación:** se considera como una unión que permite el movimiento relativo entre eslabones consecutivos, formando una cadena cinemática. Existen una gran variedad de articulaciones, entre las más utilizadas se tiene: rotacional, prismática y cilíndrica [6].
- **Grado de libertad (GDL):** es cada uno de los movimientos que puede hacer una articulación independientemente de la anterior.
- **Efecto final:** son dispositivos que interactúan directamente con el entorno del robot, y son diseñados para un tipo de trabajo en particular. Según la ISO 8373 es un “Objeto unido a la interfaz mecánica dispuesta en el extremo del robot, para dotarle a éste de la funcionalidad necesaria para realizar su tarea”. Pueden ser: elementos de sujeción o herramientas [1].
- **Cadena cinemática:** es un conjunto de eslabones, los cuales están unidos por medio de articulaciones como se muestra en la Figura 1.1. Se tiene dos tipos: [6]
 - **Cadena cinemática cerrada:** configuración en la cual se puede llegar desde cualquier eslabón a otro por medio de al menos dos caminos (Figura 1.1a).
 - **Cadena cinemática abierta:** configuración en la que se llega a un eslabón por un solo camino (Figura 1.1b).

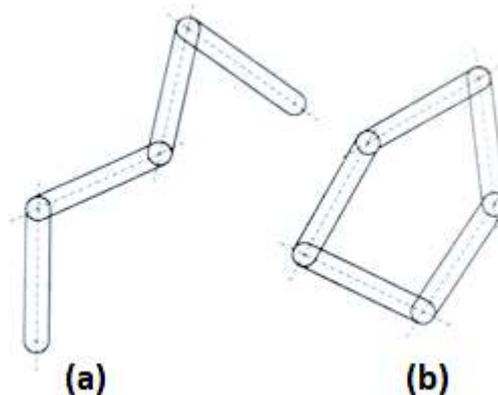


Figura 1.1. Cadena cinemática: (a) abierta y (b) cerrada.
(Fuente: Barrientos, Peñín, Balaguer & Aracil, 1997)

Desde el enfoque científico, los manipuladores presentan un amplio campo de retos tanto teóricos como prácticos debido a su naturaleza dinámica multivariable y no lineal, siendo una motivación para el desarrollo de este proyecto. Los campos de aplicación son varios, por ejemplo: en medicina, educación, tele-operación, misiones espaciales, manufactura, investigación, entre otros [1, 6, 7].

1.2.1. Configuraciones cinemáticas

La combinación de articulaciones en un robot genera diversas configuraciones, con características y aplicaciones específicas. En el caso de los manipuladores de cadena cinemática abierta, las configuraciones más frecuentes son: cartesiana, cilíndrica, polar o esférica, scara, angular o antropomórfica. Siendo esta última constituida por articulaciones de tipo angular (Figura 1.2) y una de la más usada. De ahí su importancia para este proyecto ya que presenta ventajas como: facilidad para efectuar trayectorias complejas, alta maniobrabilidad y acceso a entornos con obstrucciones [1, 2].

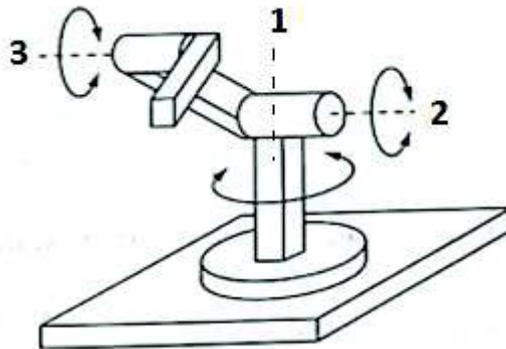


Figura 1.2. Configuración cinemática angular para un robot de 3GDL.
Articulaciones angulares giran en torno al eje 1, 2 y 3 bajo un sistema de referencia xyz.

(Fuente: Barrientos, Peñín, Balaguer & Aracil, 1997)

1.2.2. Volumen de trabajo

Es aquel que es capaz de cubrir el efector final cuando el robot realiza movimientos.

- **Espacio de trabajo alcanzable:** conjunto de puntos que puede alcanzar el robot.
- **Espacio de trabajo diestro:** conjunto de puntos que puede alcanzar el robot con una orientación arbitraria, siendo éste un subconjunto del anterior.

1.2.3. Sistemas funcionales de un manipulador

Un robot está conformado por las siguientes partes fundamentales: [1,2]

- **Mecánica:** mecánicamente un manipulador está constituido por una serie de eslabones unidos por medio de articulaciones, cuyo aspecto físico tiene una similitud con la anatomía de un brazo humano (cintura, brazo, antebrazo, codo y muñeca).
- **Actuadores:** estos elementos proporcionan las fuerzas y pares que se requieren para que un robot pueda moverse, en función de las señales generadas por un sistema de control. Las tecnologías empleadas son: hidráulica, neumática y eléctrica.
- **Sistemas de transmisión y reducción:** en los robots se requiere que los movimientos se transmitan desde los actuadores a las articulaciones, buscando adaptar el par y la velocidad del actuador para que el robot realice movimientos

adecuados (reductores por engranes, tornillos de bolas, reductores planetarios, reductores armónicos, bandas, cadenas, accionamientos directos, entre otros).

- **Sensores:** En el campo de la robótica uno de los objetivos es que un robot realice tareas con precisión, velocidad adecuada y cierto grado de inteligencia, por lo cual es necesario que tenga la capacidad para obtener información de su propio estado y del entorno (sensores internos y entornos externos).
- **Sistema de control:** De manera general un controlador puede considerarse como un sistema computacional que, en conjunto con algoritmos de control, realizan cálculos con el fin de que el robot realice movimientos acordes a la actividad para la cual fue diseñado. Las señales de control generadas deben ser amplificadas en una etapa de potencia para comandar los actuadores.

1.3. Robots operados remotamente

Desde tiempos antiguos, el ser humano ha buscado usar herramientas que le permitan extender su capacidad de manipulación, así como trabajar con sustancias peligrosas. Además, en varias aplicaciones de robótica la intervención del ser humano es imprescindible, sobre todo en entornos muy dinámicos en donde la planificación automática puede ser muy complicada. Es así como surge el campo de la tele-operación, es decir sistemas (robots) controlados remotamente. Este campo ha tenido un gran desarrollo, pues ha permitido la integración de áreas como: robótica, electrónica, comunicaciones, control, inteligencia artificial, entre otros [4, 20].

Desde el enfoque de la robótica de manipuladores, un sistema operado remotamente generalmente consiste en un manipulador esclavo que reproduce los movimientos realizados desde un sistema de control o manipulador maestro, el cual es controlado a su vez por un operador humano. Así mismo en este campo puede entenderse como tele-operación a *“la extensión de capacidades sensoriales y destreza humana a una localización remota”* [4].

Con estos sistemas se busca brindar seguridad al usuario, sobre todo si se van a realizar trabajos en ambientes inseguros. La intervención del operador puede producirse en muchas formas diferentes, desde la tele-operación directa de actuadores de un robot, hasta la especificación de movimientos o tareas a desarrollarse de manera automática. En la mayoría de los usos de estos sistemas el operador está físicamente alejado del robot, siendo necesario la existencia de un sistema de telecomunicaciones entre la interfaz de control y el sistema de control local del robot.

En base a lo mencionado, un sistema de tele-operación está conformado por:

- **Operador o teleoperador:** es un ser humano que realiza un control a distancia, el cual puede ser continuo o intermitente.
- **Dispositivo tele-operado (esclavo):** es un sistema (manipulador, vehículo móvil) ubicado en una zona remota, el cual es controlado por el operador. Generalmente está provisto por sensores que dan a conocer el entorno al operador, brindando una sensación de estar en dicho lugar (tele-presencia). También está constituido por una unidad de procesamiento en donde se almacenan: algoritmos de control, así como un programa el cual le da cierto grado de autonomía en circunstancias en las cuales el canal de comunicación falle.
- **Estación de control (Interfaz):** es un conjunto de dispositivos y/o algoritmos de control que permiten mandar y recibir información entre el operador y el sistema esclavo. Entre estos están: un computador, monitor de video, joystick, guantes, exoesqueletos, etc. Además, se puede disponer de una interfaz hombre-máquina la cual debe estar diseñada de tal manera que facilite al usuario obtener la información esencial, así como una adecuada ubicación espacial del robot.
- **Sistema de comunicación:** es el medio por el cual se pueden enviar las señales de control desde la estación remota hacia el esclavo, y viceversa. Este canal de comunicación puede darse por medio de una red local, enlace de radio frecuencia, y actualmente internet/nube.

Actualmente este campo de estudio ha generado grandes retos que han sido de gran interés para investigadores, de los cuales se podrían mencionar:

- Compensación entre las cinemáticas del esclavo y la interfaz de control.
- Algoritmos para planeación de trayectorias por ejemplo para el traslado de objetos.
- Procesamiento de información en la estación de control bajo restricciones de tiempo para la toma de acciones inmediatas.
- Sistemas predictivos para mejorar la eficiencia con respecto a retardos de tiempo.
- Compensación de retardo en la comunicación.
- Técnicas de inteligencia artificial para mayor autonomía, por ejemplo, en la evasión de obstáculos no tomados en cuenta por el operador.
- Sistemas con realimentación de esfuerzos (hápticos), entre otros.

Debido a los varios campos de aplicación, este proyecto busca incursionar en esos sistemas mediante la operación remota de un robot virtual, por lo cual se emplearán las terminologías: sistema esclavo (robot virtual y algoritmos de control), interfaz de control (brazaletes y algoritmos de reconocimiento de gestos) y sistema de comunicación (red WLAN).

1.4. Plataforma robótica seleccionada

En el mercado industrial existen compañías que proporcionan manipuladores muy precisos y potentes, ofreciendo grandes prestaciones a costos elevados. Esto es un inconveniente para ser usados en pequeñas empresas, o en temas de investigación, por lo que ha generado una necesidad de abrir nuevos mercados que brinden opciones de bajo costo. Actualmente se pueden encontrar una variedad de brazos robóticos a costos accesibles, siendo muy usados en temas de investigación y aplicaciones robóticas.

El proyecto busca el control de un manipulador de 3GDL en un entorno virtual, para lo cual se consideró manipuladores robóticos de configuración angular de bajo costo. Es así como se realizó una indagación de mercado, pudiéndose encontrar estructuras de hardware abierto y comerciales de bajo costo. En la Tabla 1.1, se muestran 4 manipuladores con sus características más relevantes.

Tabla 1.1. Manipuladores robóticos.

Característica	Manipulador			
	Thor	Niryo One	Lynxmotion AL5D	PhantomX Reactor
GDL	5 + pinza	5 + pinza	4 + pinza	5 + pinza
Carga máxima	0.75 kg	0.5 kg	0.28 kg	0.25 Kg
Alcance Máximo	62.5 cm	44 cm	19 cm	38 cm
Costo	No tiene costo	\$1140	\$310.81	\$600
Material	Plástico ABS	Aluminio PLA-3D printing	Aluminio	Delrin, acrílico y metal
WEB	www.thingiverse.com	www.niryo.com	www.lynxmotion.com	www.trossenrobotics.com
Imagen referencial				

(Fuente: Propia)

Para la selección de la estructura usada en este trabajo, se ha empleado el método de matrices de ponderación, la cual ha permitido seleccionar la alternativa con características que se adapta para el cumplimiento del objetivo principal del proyecto.

En la Tabla 1.2 se puede apreciar que el parámetro más relevante es que el fabricante de la alternativa a seleccionar proporcione información paramétrica (dimensiones, formas y material) del robot para su modelamiento CAD.

Tabla 1.2. Criterios y ponderaciones para selección.

Criterio	Descripción	Ponderación
1	Alcance máximo del manipulador robótico.	0.2
2	Capacidad de carga máxima del manipulador.	0.1
3	Sistema modular expansión y modificación.	0.2
4	Grados de libertad del robot.	0.05
5	Material resistente y liviano.	0.2
6	Información sobre dimensiones de partes y parámetros del robot.	0.25
	Total	1

(Fuente: Propia)

Por lo tanto, la matriz de selección desarrollada se muestra en la Tabla 1.3, en la cual se consideró un puntaje de 0 a 5 para cada criterio.

Tabla 1.3. Matriz para selección del Manipulador.

Criterio	Peso	Alternativa 1: Thor		Alternativa 2: Niryo One		Alternativa 3: Lynxmotion AL5D		Alternativa 4: PhantomX Reactor	
1	0.2	5	1	4	0.8	2	0.4	3	0.6
2	0.1	5	0.5	4	0.4	3	0.3	2	0.2
3	0.2	0	0	0	0	5	1	4	0.8
4	0.05	5	0.3	5	0.3	4	0.2	5	0.3
5	0.2	2	0.4	3	0.6	4	0.8	5	1
6	0.25	4	1	1	0.3	5	1.3	2	0.5
		Total	3.2	Total	2.3	Total	4	Total	3.4

(Fuente: Propia)

De la Tabla 1.3, se aprecia que la mejor alternativa es la opción 3, correspondiente al manipulador del fabricante Lynxmotion, modelo AL5D, la cual está conformada por los siguientes elementos:

- Cuatro servomotores Hitec: HS-485HB para la base; HS-805BB para el hombro; HS-755 para el codo; HS-645MG para la muñeca; HS-422 para la pinza.
- Una base giratoria con rodamientos tipo esfera BR-NS.
- Tres soportes multipropósito para servomotor (ASB-204, ASB-201, ASB-04)
- Dos soportes largo tipo C para servomotor (ASB-203, ASB-205), con dos paneles de amortiguamiento tipo disco (MD-01)
- Dos soportes conectores tipo L (ASB-06).
- Dos conectores de tubo central (HUB-08).
- Un tubo principal (AT-03) y una pinza.

1.5. Modelamiento CAD 3D de la plataforma robótica

El proyecto no involucra considerar las piezas de la muñeca y de la pinza del robot comercial. Es así como se realizó la modelación y ensamble del manipulador empleando el software Autodesk Inventor 2018, en este caso se ha empleado la información técnica (formas, dimensiones, materiales de piezas) suministrada por el fabricante del robot (disponible en: www.lynxmotion.com/).

1.5.1. Elementos modelados

Base giratoria BR-NS

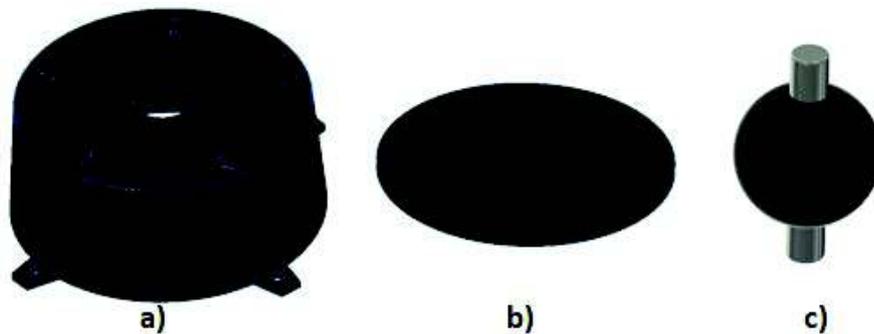


Figura 1.3. a) Base fija; b) Tapa giratoria; c) Rodamiento.
(Fuente: Propia)

Soporte multipropósito para servomotor

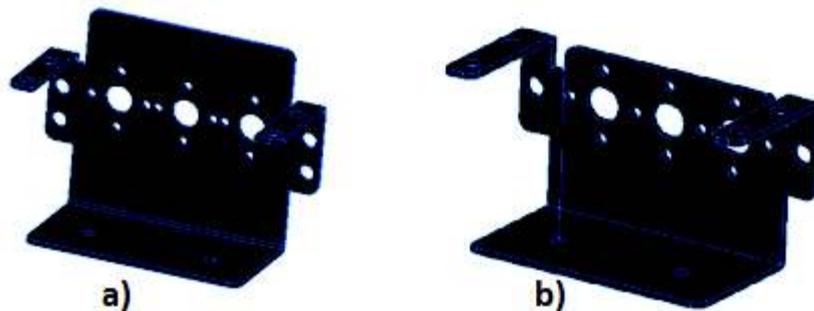


Figura 1.4. Soportes multipropósito para servomotor: a) ASB204; b) ASB201.
(Fuente: Propia)

Soporte tipo C

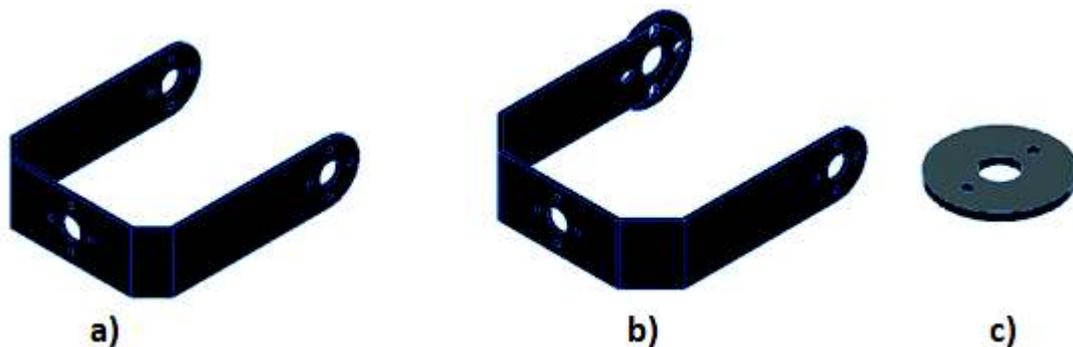


Figura 1.5. Soportes largo tipo C para servomotor: a) ASB203; b) ASB205 y c) Paneles tipo disco.
(Fuente: Propia)

Tubo principal y soportes

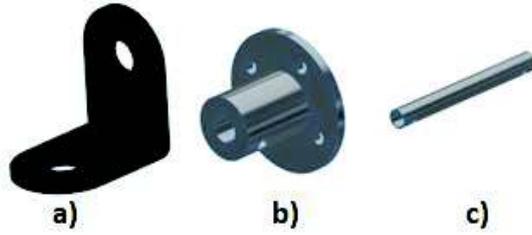


Figura 1.6. a) Soporte tipo L; b) Conector; c) Tubo central.
(Fuente: Propia)

Servomotores

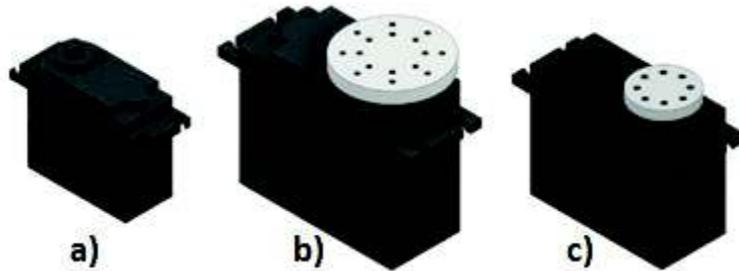


Figura 1.7. Servomotores: a) HS-485HB, b) HS-805BB, c) HS-755.
(Fuente: Propia)

1.5.2. Ensamblajes

Mediante las restricciones correspondientes, se realizó el ensamble de los 4 eslabones del robot (Base fija, Hombro, Brazo y Antebrazo):

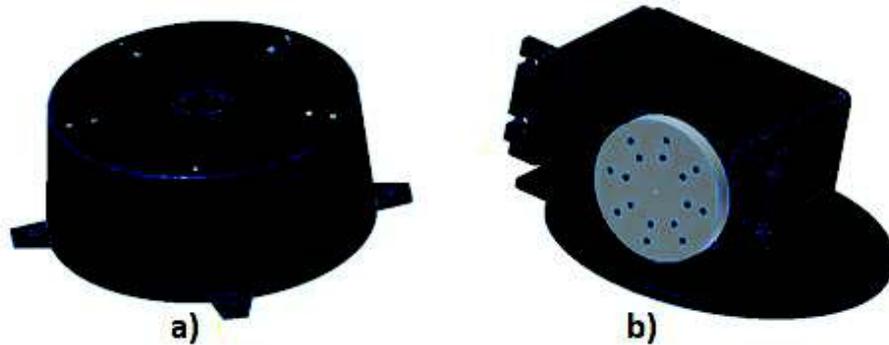


Figura 1.8. a) Base fija; b) Hombro.
(Fuente: Propia)

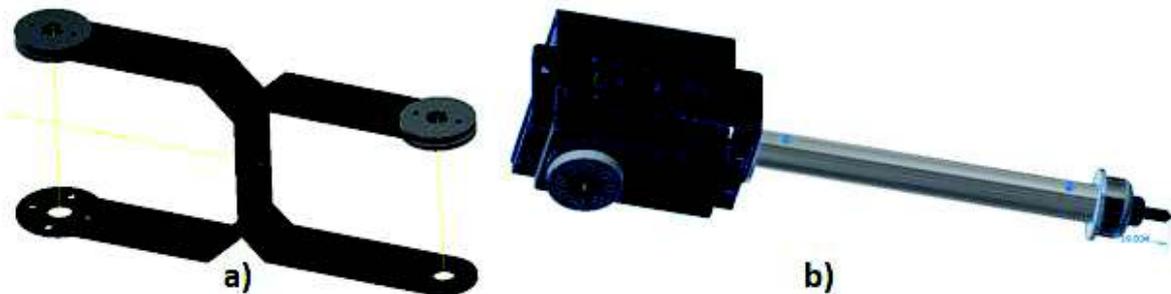


Figura 1.9. a) Brazo; b) Antebrazo.
(Fuente: Propia)

El robot modelado en 3D tiene una configuración antropomórfica con 3GDL y articulaciones rotacionales, y los ejes de giro de las articulaciones se muestran en la Figura 1.10.

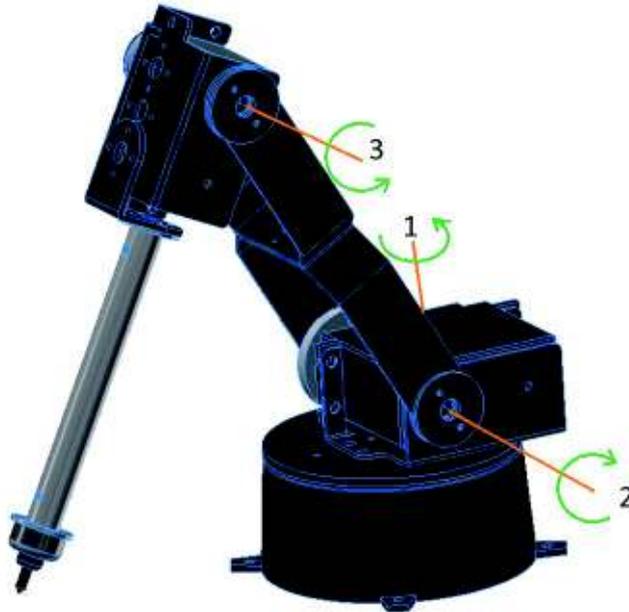


Figura 1.10. Manipulador robótico 3GDL.
Los ejes 1, 2 y 3 permiten la rotación de las articulaciones 1, 2 y 3 respectivamente.
(Fuente: Propia)

1.5.3. Espacio de trabajo

En la Figura 1.11, considerando las limitaciones angulares y dimensiones de los eslabones, se muestra el espacio de trabajo alcanzable y diestro (sombreado) en milímetros (mm) de la plataforma robótica seleccionada.

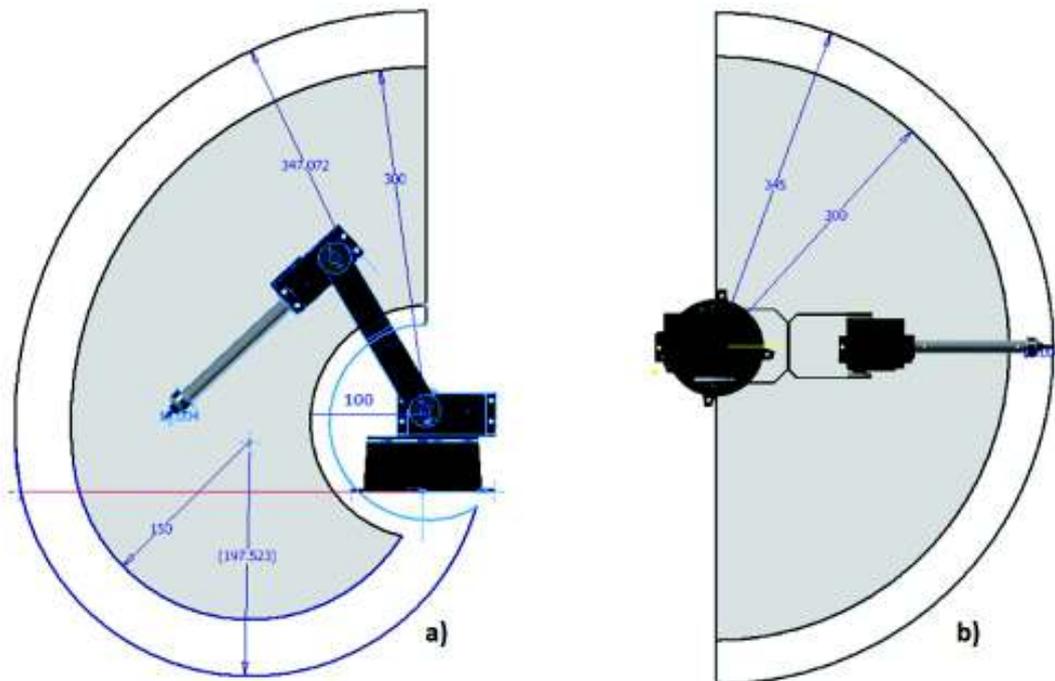


Figura 1.11. Vistas: a) lateral izquierda y b) superior, presentadas con dimensiones en milímetros.
(Fuente: Propia)

Generalmente las limitaciones angulares están establecidas de acuerdo con el rango de trabajo de los actuadores de la plataforma comercial (generalmente 180°), y a requerimientos del espacio de trabajo, por lo que para el proyecto se ha considerado las limitaciones angulares mostradas en la Tabla 1.4.

Tabla 1.4. Limitaciones angulares.

Articulación	1	2	3
Rango	-90° a 90°	0° a 180°	-150° a 0°

(Fuente: Propia)

Donde las articulaciones 1, 2 y 3 hacen referencia a los movimientos rotacionales de la base del robot, brazo y antebrazo respectivamente. Además, de la Tabla 1.4 un ángulo positivo se considera en sentido antihorario, y un ángulo negativo en sentido horario.

1.5.4. Parámetros del manipulador

Con referencia a los datos técnicos del fabricante (www.lynxmotion.com) en el cual se indica el tipo de material (plástico ABS y aluminio adonizado), se obtuvieron los parámetros aproximados de dimensiones, masa, inercias y ubicación de los centros de masa (Tabla 1.5), esto gracias al uso de herramientas del software Autodesk Inventor 2018. Dichos parámetros son usados en los estudios cinemático y dinámico como se explicará más adelante.

Tabla 1.5. Parámetros del manipulador virtual.

	Hombro	Brazo (L1)	Antebrazo (L2)
Longitud (m)	-	0.149607	0.201537
Masa (Kg)	m1= 0.198	m2= 0.046	m3= 0.258
Centro de masa – CM (m)	x1=0.017158; y1=-0.003144; z1=-0.00407	x2=0.073995; y2=0; z2=0	x3= 0.088263; y3=0; z3=0
Tensor de inercia del CM (Kg.mm ²)	$I_{c1} = \begin{bmatrix} 97.6 & 0 & 0 \\ 0 & 112.7 & 0 \\ 0 & 0 & 158.3 \end{bmatrix}$	$I_{c2} = \begin{bmatrix} 86.8 & 0 & 0 \\ 0 & 47.6 & 0 \\ 0 & 0 & 129.6 \end{bmatrix}$	$I_{c3} = \begin{bmatrix} 50.1 & 0 & 0 \\ 0 & 1709.1 & 0 \\ 0 & 0 & 1728.9 \end{bmatrix}$

(Fuente: Propia)

Donde:

- L_1 y L_2 : son las longitudes de los eslabones (2 y 3) brazo y antebrazo, respectivamente.
- m_1 , m_2 y m_3 : son las masas de los eslabones (1, 2 y 3) hombro, brazo y antebrazo, respectivamente.
- x_i , y_i y z_i : son las coordenadas cartesianas del centro de masa (CM) del eslabón i con respecto a la articulación i , donde $i=1; 2; 3$.

En base a lo detallado con respecto a la plataforma robótica, a continuación, se presenta el estudio cinemático y dinámico del manipulador. Es importante mencionar que todas las operaciones han sido realizadas usando el software Matlab R2018a.

1.6. Cinemática de manipuladores

Es un estudio que corresponde al análisis del movimiento del manipulador con respecto a un sistema de referencia, sin considerar las fuerzas que producen dichos movimientos [6,7]. De esta manera, en función de los 3 GDL, se define el vector de coordenadas articulares generalizadas o vector de posiciones articulares, $\mathbf{q} = [q_1 \ q_2 \ q_3]^T \in R^3$, que por medio de las propiedades geométricas del manipulador se busca establecer una relación con las coordenadas cartesianas $[x, y, z]^T \in R^3$ y de orientación.

En este estudio es necesario la aplicación de conceptos como: rotaciones, traslaciones y matriz de transformación homogénea [6]. Además, aquí se tienen dos problemas fundamentales: cinemática directa y cinemática inversa, los cuales para el manipulador de este proyecto se resuelven en los siguientes puntos.

1.6.1. Problema de la cinemática directa

Aquí se determina la posición y orientación del extremo del robot, con respecto a una referencia fija en la base inmóvil del robot (Ecuación 1.1).

$$[x \ y \ z \ \alpha \ \beta \ \gamma]^T = f(\mathbf{q}) \quad (1.1)$$

La metodología empleada es la de Denavit-Hartenberg (DH) [6], la cual, mediante una serie de transformaciones entre eslabones, establece una matriz de transformación homogénea constituida por ecuaciones cinemáticas del extremo final referenciado a la base fija del manipulador.

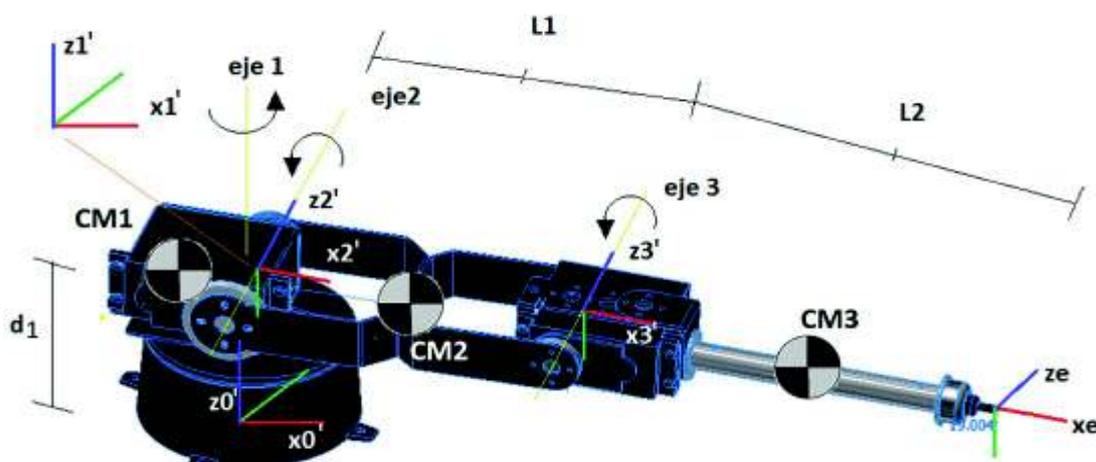


Figura 1.12. Asignación de sistemas de referencia. En este caso se considera: la base fija como el eslabón 0, al hombro como el eslabón 1, al brazo como el eslabón 2 y al antebrazo como el eslabón 3. (Fuente: Propia)

La aplicación de este método inicia con la identificación de eslabones, articulaciones. Posteriormente se ubican los sistemas de referencia en cada articulación tal como se muestra en la Figura 1.12.

Donde:

- x'_0, y'_0 y z'_0 : son los ejes del sistema de referencia fijo ubicado en la base del manipulador.
- x'_i, y'_i y z'_i : son los ejes del sistema de referencia ubicado en la articulación i (1, 2 y 3).
- x_e y z_e son los ejes del sistema de referencia del extremo del robot.
- CM_i corresponde a la ubicación del centro de masa del eslabón i (1, 2 y 3).
- d_1 es la altura desde el sistema de referencia de la articulación 1 al sistema de referencia 0 ($d_1=0.0675m$).

Dentro de este método se definen 4 cantidades denotadas como parámetros DH y relacionan los sistemas de referencia de dos eslabones consecutivos [6].

- **Parámetro α_{i-1}** : distancia entre los ejes z_{i-1} y z_i (ejes i e $i-1$), a lo largo de x_{i-1} , la cual representa la longitud del eslabón.
- **Parámetro α_{i-1}** : ángulo entre los ejes z_{i-1} y z_i a lo largo de x_{i-1} , es decir define la rotación entre los eslabones desde $i-1$ a i .
- **Parámetro d_i** : es la distancia con signo desde el eje x_{i-1} al eje x_i a lo largo de z_i (offset).
- **Parámetro θ_i** : ángulo entre los ejes x_{i-1} y x_i a lo largo del eje z_i a lo largo de x_{i-1} . En el caso de una articulación rotacional éste es un parámetro variable.

Conforme a la descripción de estos parámetros, en la Tabla 1.6 se muestran los parámetros del manipulador modelado en este proyecto.

Tabla 1.6. Parámetros DH.

i	α_{i-1} (rad)	a_{i-1}	d_i	θ_i (rad)
1	0	0	d_1	θ_1
2	$\pi/2$	0	0	θ_2
3	0	L_1	0	θ_3
e	0	L_2	0	0

(Fuente: Propia)

Donde θ_i corresponde al ángulo de giro de la articulación i (1, 2 y 3). Además, estos ángulos corresponden a los términos del vector $\mathbf{q} = [q_1 \ q_2 \ q_3]^T$.

Además, los parámetros DH permiten establecer la matriz de transformación homogénea que relaciona el eje de articulación $i-1$ e i :

$$T_i^{i-1} = \begin{bmatrix} \cos(\theta_i) & -\text{sen}(\theta_i) & 0 & a_{i-1} \\ \text{sen}(\theta_i) \cos(\alpha_{i-1}) & \cos(\theta_i) \cos(\alpha_{i-1}) & -\text{sen}(\alpha_{i-1}) & -d_i \text{sen}(\alpha_{i-1}) \\ \text{sen}(\theta_i) \text{sen}(\alpha_{i-1}) & \cos(\theta_i) \text{sen}(\alpha_{i-1}) & \cos(\alpha_{i-1}) & d_i \cos(\alpha_{i-1}) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.2)$$

De esta manera, usando la Ecuación 1.2, se obtienen las matrices de transformación en cada una de las articulaciones y del extremo del robot:

$$T_1^0 = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}; T_2^1 = \begin{bmatrix} c_2 & -s_2 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s_2 & c_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.3)$$

$$T_3^2 = \begin{bmatrix} c_3 & -s_3 & 0 & L_1 \\ s_3 & c_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; T_e^3 = \begin{bmatrix} 1 & 0 & 0 & L_2 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Por lo tanto, las matrices de transformación de las articulaciones 2 y 3, además del efector final con respecto a la base (sistema de referencia 0) del manipulador son:

$$T_2^0 = T_1^0 T_2^1 = \begin{bmatrix} c_1 c_2 & -c_1 s_2 & s_1 & 0 \\ s_1 c_2 & -s_1 s_2 & -c_1 & 0 \\ s_2 & c_2 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.4)$$

$$T_3^0 = T_1^0 T_2^1 T_3^2 = \begin{bmatrix} c_1 c_{23} & -c_1 s_{23} & s_1 & c_1 c_2 L_1 \\ s_1 c_{23} & -s_1 s_{23} & -c_1 & s_1 c_2 L_1 \\ s_{23} & c_{23} & 0 & d_1 + s_2 L_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.5)$$

$$T_e^0 = T_1^0 T_2^1 T_3^2 T_e^3 = \begin{bmatrix} c_1 c_{23} & -c_1 s_{23} & s_1 & c_1 c_2 L_1 + c_1 c_{23} L_2 \\ s_1 c_{23} & -s_1 s_{23} & -c_1 & s_1 c_2 L_1 + s_1 c_{23} L_2 \\ s_{23} & c_{23} & 0 & d_1 + s_2 L_1 + s_{23} L_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.6)$$

Finalmente, las coordenadas en el espacio cartesiano del extremo del robot son:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} c_1 c_2 L_1 + c_1 c_{23} L_2 \\ s_1 c_2 L_1 + s_1 c_{23} L_2 \\ d_1 + s_2 L_1 + s_{23} L_2 \end{bmatrix} \quad (1.7)$$

Es importante mencionar que se ha empleado la nomenclatura donde:

- s_i y c_i corresponden a $\text{sen}(\theta_i)$ y $\cos(\theta_i)$ respectivamente.
- s_{ij} y c_{ij} corresponden a $\text{sen}(\theta_i + \theta_j)$ y $\cos(\theta_i + \theta_j)$ respectivamente.

1.6.2. Problema de la cinemática inversa

Se trata de la determinación de los valores articulares a partir de una posición específica del manipulador en el sistema cartesiano, es decir lo contrario de la cinemática directa.

$$\mathbf{q} = f(x, y, z) \quad (1.8)$$

De lo mencionado, en el proyecto se quiere controlar el movimiento del robot usando datos de las IMU de los brazaletes, por lo que usando datos de orientación de las IMU mientras el usuario realiza movimientos, se quiere establecer un punto en el espacio de trabajo del robot, y mediante la resolución de la cinemática inversa obtener los valores de las articulaciones del robot que generan dicho punto.

Este problema implica la operación con ecuaciones no lineales, las cuales pueden tener varias soluciones o no tener solución. La existencia de múltiples soluciones significa que el manipulador tiene muchas configuraciones para llegar a la posición y orientación deseada, mientras que, en el caso de no haber solución, significa que la posición y orientación dada es imposible para el manipulador ya que el punto estaría fuera del espacio de trabajo descrito en la Sección 1.4.4.

Existen varias técnicas que permiten resolver este problema [6], clasificándose en:

- **En forma cerrada:** proporcionan el valor deseado de forma directa.
- **Numéricas:** usan métodos iterativos para determinar el valor de una variable.

Por lo cual, se ha considerado la técnica en forma cerrada, pues permite la resolución en tiempo real, permitiendo incluir restricciones para la obtención de una solución adecuada, además en trabajos futuros sería útil por ejemplo en el seguimiento de trayectoria. El método optado dentro de esta técnica es el geométrico pues es sencillo de usar debido a los pocos grados de libertad que tiene el manipulador. El mismo se sustenta en la obtención de relaciones geométricas en las que aparecen las dimensiones del robot, coordenadas de las articulaciones y las coordenadas del extremo del robot.

En el Capítulo 2 se detallará el desarrollo de este método para el robot en función de los movimientos seleccionados desde la interfaz de control.

1.6.3. Jacobiano

El Jacobiano J del manipulador, es una herramienta matemática que relaciona las velocidades articulares con la velocidades lineales y angulares del efector final. El Jacobiano permite una obtención más directa de las ecuaciones de movimiento con el método de Euler-Lagrange [6], de ahí su importancia de su obtención en este trabajo.

Es así que se ha considerado el método directo para el cálculo del Jacobiano, siendo su representación general:

$$\begin{bmatrix} v \\ w \end{bmatrix} = J(\mathbf{q})\dot{\mathbf{q}} = \begin{bmatrix} J_v \\ J_w \end{bmatrix} \dot{\mathbf{q}} \quad (1.9)$$

Donde $\dot{\mathbf{q}}$ es el vector de velocidad articular cuya dimensión es de 3×1 , $\dot{\mathbf{q}} = [\dot{q}_1 \ \dot{q}_2 \ \dot{q}_3]^T$.

La sub-matriz J_v relaciona las velocidades articulares $\dot{\mathbf{q}}$ con las velocidades lineales del extremo del robot v .

$$v = J_v \dot{\mathbf{q}} \quad (1.10)$$

Por lo que considerando el vector posición de cada centro de masa de la Figura 1.12 y en base a lo obtenido en la Sección 1.5.1, se tiene:

$$\begin{bmatrix} P_{CM1}^0 \\ 1 \end{bmatrix} = T_1^0 \begin{bmatrix} x_1 \\ y_1 \\ z_1 \\ 1 \end{bmatrix} = \begin{bmatrix} c_1 x_1 - s_1 y_1 \\ s_1 x_1 + c_1 y_1 \\ d_1 + z_1 \\ 1 \end{bmatrix} \quad (1.11)$$

$$\begin{bmatrix} P_{CM2}^0 \\ 1 \end{bmatrix} = T_2^0 \begin{bmatrix} x_2 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} c_1 c_2 x_2 \\ c_2 s_1 x_2 \\ d_1 + s_2 x_2 \\ 1 \end{bmatrix} \quad (1.12)$$

$$\begin{bmatrix} P_{CM3}^0 \\ 1 \end{bmatrix} = T_3^0 \begin{bmatrix} x_3 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} c_1 c_{23} x_3 + c_1 c_2 L_1 \\ c_{23} s_1 x_3 + s_1 c_2 L_1 \\ d_1 + s_2 L_1 + s_{23} x_3 \\ 1 \end{bmatrix} \quad (1.13)$$

De manera que el J_v , se determina derivando los vectores de las Ecuaciones 1.11, 1.12 y 1.13 con respecto al vector \mathbf{q} , obteniendo:

$$Jv_{c1} = \begin{bmatrix} -s_1 x_1 - c_1 y_1 & 0 & 0 \\ c_1 x_1 - s_1 y_1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (1.14)$$

$$Jv_{c2} = \begin{bmatrix} -s_1 c_2 x_2 & -c_1 s_2 x_2 & 0 \\ c_1 c_2 x_2 & -s_2 s_1 x_2 & 0 \\ 0 & c_2 x_2 & 0 \end{bmatrix} \quad (1.15)$$

$$Jv_{c3} = \begin{bmatrix} -s_1 c_{23} x_3 - s_1 c_2 L_1 & -c_1 s_{23} x_3 - c_1 s_2 L_1 & -c_1 s_{23} x_3 \\ c_{23} c_1 x_3 + c_1 c_2 L_1 & -s_{23} s_1 x_3 - s_1 s_2 L_1 & -s_{23} s_1 x_3 \\ 0 & c_2 L_1 + c_{23} x_3 & c_{23} x_3 \end{bmatrix} \quad (1.16)$$

La sub-matriz J_w relaciona las velocidades articulares $\dot{\mathbf{q}}$ con las velocidades angulares del extremo del robot w .

$$w = J_w \dot{\mathbf{q}} \quad (1.17)$$

Para determinar J_w , se considera la rotación R del eje de acción de las articulaciones con respecto al sistema de referencia 0 (base fija), es decir, $z_i^0 = R_i^0 z$, con $z = [0 \ 0 \ 1]^T$, ya que en base al método DH cada eslabón gira en torno al eje z_i' . La matriz de rotación es una sub matriz de las transformadas homogéneas obtenidas en la Sección 1.5.1

$$z_1^0 = R_1^0 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (1.18)$$

$$z_2^0 = R_2^0 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} s_1 \\ -c_1 \\ 1 \end{bmatrix} \quad (1.19)$$

$$z_3^0 = R_3^0 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} s_1 \\ -c_1 \\ 1 \end{bmatrix} \quad (1.20)$$

Finalmente, en base a lo obtenido en las Ecuaciones 1.18, 1.19 y 1.20, se obtiene J_w de cada articulación con respecto a la base fija: [6]

$$Jw_1^0 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad (1.21)$$

$$Jw_2^0 = \begin{bmatrix} 0 & s_1 & 0 \\ 0 & -c_1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad (1.22)$$

$$Jw_3^0 = \begin{bmatrix} 0 & s_1 & s_1 \\ 0 & -c_1 & -c_1 \\ 1 & 0 & 0 \end{bmatrix} \quad (1.23)$$

1.7. Dinámica del manipulador

De lo visto anteriormente, la cinemática no considera las fuerzas que causan el movimiento de un manipulador. Es así como en el caso de la dinámica se busca obtener y estudiar las ecuaciones que describen el movimiento y comportamiento físico de los mecanismos, enfocándose en que estos cada vez se aproximen al comportamiento real. Estos modelos son un conjunto de ecuaciones diferenciales que relacionan los fenómenos físicos involucrados en los movimientos de un robot, siendo también conocidos como ecuaciones de movimiento [6, 7, 11].

Existen algunos métodos para determinar dichas ecuaciones, como: Newton-Euler y Euler-Lagrange (EL), siendo este último muy utilizado en manipuladores ya que permite la obtención directa de dichas ecuaciones, y es el método adoptado en este trabajo.

En este método se define el Lagrangiano L como la diferencia entre la energía cinética T y la energía potencial U del sistema.

$$L = T - U \quad (1.24)$$

Por lo que la expresión para determinar las ecuaciones EL es:

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = \tau_i; \quad i = 1, 2, 3 \quad (1.25)$$

Donde τ_i , corresponde al torque de la articulación i del manipulador.

1.7.1. Obtención de las ecuaciones dinámicas del manipulador

Como primer punto, se determina la energía cinética total del manipulador, que es igual a la sumatoria de la energía cinética T_i de cada eslabón. Siendo ésta a su vez la suma de la energía cinética generada por el movimiento traslacional T_{vi} y rotacional T_{wi} .

$$T(\mathbf{q}, \dot{\mathbf{q}}) = \sum_{i=1}^3 T_i = \sum_{i=1}^3 (T_{vi} + T_{wi}) = \frac{1}{2} \sum_{i=1}^3 [m_i v_{ci}^T v_{ci} + w_i^T I_{ci} w_i] \quad (1.26)$$

Donde v_{ci} e I_{ci} son la velocidad lineal y la inercia del centro de masa el i -ésimo eslabón, respectivamente. Aplicando el concepto del Jacobiano a la Ecuación 1.26, se tiene:

$$T(\mathbf{q}, \dot{\mathbf{q}}) = \frac{1}{2} \dot{\mathbf{q}}^T \left\{ \sum_i [m_i J_{vi}^T J_{vi} + J_{wi}^T I_{ci} J_{wi}] \right\} \dot{\mathbf{q}} \quad (1.27)$$

De manera que en base a la Ecuación 1.27, se obtiene T_v y T_w .

$$T_v = \frac{1}{2} \dot{\mathbf{q}}^T [m_1 [J_{v_{c1}^0}]^T J_{v_{c1}^0} + m_2 [J_{v_{c2}^0}]^T J_{v_{c2}^0} + m_3 [J_{v_{c3}^0}]^T J_{v_{c3}^0}] \dot{\mathbf{q}} \quad (1.28)$$

$$T_w = \frac{1}{2} \dot{\mathbf{q}}^T [J_{w_1^1}]^T I_{c1} J_{w_1^1} + [J_{w_2^2}]^T I_{c2} J_{w_2^2} + [J_{w_3^3}]^T I_{c3} J_{w_3^3}] \dot{\mathbf{q}} \quad (1.29)$$

Con la finalidad de realizar simplificaciones en el proceso de obtención de las ecuaciones dinámicas, se han considerado variables simbólicas I_{xxi} , I_{yyi} , I_{zz1} , las cuales representan los elementos de la diagonal del tensor de inercia del centro de masa i .

$$I_{C1} = \begin{bmatrix} I_{xx1} & 0 & 0 \\ 0 & I_{yy1} & 0 \\ 0 & 0 & I_{zz1} \end{bmatrix}; \quad I_{C2} = \begin{bmatrix} I_{xx2} & 0 & 0 \\ 0 & I_{yy2} & 0 \\ 0 & 0 & I_{zz2} \end{bmatrix}; \quad I_{C3} = \begin{bmatrix} I_{xx3} & 0 & 0 \\ 0 & I_{yy3} & 0 \\ 0 & 0 & I_{zz3} \end{bmatrix} \quad (1.30)$$

Además, en base a lo obtenido en las Ecuaciones 1.21, 1.22 y 1.23, se tiene que:

$$J_{w_1^1} = [R_1^0]^T J_{w_1^0} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}; \quad (1.31)$$

$$J_{w_2^2} = [R_2^0]^T J_{w_2^0} = \begin{bmatrix} s_2 & 0 & 0 \\ c_2 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}; \quad J_{w_3^3} = [R_3^0]^T J_{w_3^0} = \begin{bmatrix} s_{23} & 0 & 0 \\ c_{23} & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix}$$

Finalmente empleando la Ecuación 1.27, la energía cinética del manipulador queda:

$$T(\mathbf{q}, \dot{\mathbf{q}}) = \frac{1}{2} \dot{\mathbf{q}}^T M(\mathbf{q}) \dot{\mathbf{q}} \quad (1.32)$$

Donde $M(\mathbf{q}) \in R^{3 \times 3}$ es la matriz de masa del manipulador.

$$M(\mathbf{q}) = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix}$$

$$\begin{aligned} m_{11} &= c_2^2(m_3 L_1^2 + m_2 x_2^2) + 2m_3 c_2 c_{23} L_1 x_3 + m_3 c_{23}^2 x_3^2 + m_1(x_1^2 + y_1^2) \\ &\quad + I_{yy2} c_2^2 + I_{yy3} c_{23}^2 + I_{xx2} s_2^2 + I_{xx3} s_{23}^2 + I_{zz1} \\ m_{12} &= m_{13} = m_{21} = m_{31} = 0; \end{aligned} \quad (1.33)$$

$$m_{22} = m_3 L_1^2 + 2m_3 c_3 L_1 x_3 + m_2 x_2^2 + m_3 x_3^2 + I_{zz2} + I_{zz3}$$

$$m_{23} = m_{32} = m_3 x_3(x_3 + c_3 L_1) + I_{zz3}$$

$$m_{33} = m_3 x_3^2 + I_{zz3}$$

Como siguiente paso, se obtiene la energía potencial del manipulador, la cual es la sumatoria de la energía potencial de cada eslabón U_i .

$$U(\mathbf{q}) = \sum_{i=1}^3 U_i(\mathbf{q}) \quad (1.34)$$

En base al vector de gravedad y el vector posición del centro de masa P_{ci} , que a su vez representa la altura h_i del mismo con respecto a la base del manipulador, se puede determinar la energía potencia de cada eslabón, como:

$$U_i(\mathbf{q}) = m_i [-P_{ci}^T \mathbf{g}] \quad (1.35)$$

En base a las Ecuaciones 1.34 y 1.35 y considerando el vector de gravedad $\mathbf{g} = [0 \ 0 \ -g]^T$, se obtiene:

$$\begin{aligned} U(\mathbf{q}) &= -m_1 [-P_{c1}^0{}^T \mathbf{g}] - m_2 [-P_{c2}^0{}^T \mathbf{g}] - m_3 [-P_{c3}^0{}^T \mathbf{g}] \\ U(\mathbf{q}) &= m_1 g(d_1 + z_1) + m_2 g(d_1 + s_2 x_2) + m_3 g(d_1 + s_2 L_1 + s_{23} x_3) \end{aligned} \quad (1.36)$$

En base a estas dos energías y la relación expuesta en la Ecuación 1.24, se puede determinar el Langrangiano del manipulador como:

$$\begin{aligned} L(\mathbf{q}, \dot{\mathbf{q}}) &= \frac{1}{2} [m_{11} \dot{q}_1^2 + m_{22} \dot{q}_2^2 + m_{33} \dot{q}_3^2 + 2m_{23} \dot{q}_2 \dot{q}_3] - m_1 g(d_1 + z_1) \\ &\quad - m_2 g(d_1 + s_2 x_2) - m_3 g(d_1 + s_2 L_1 + s_{23} x_3) \end{aligned} \quad (1.37)$$

Finalmente haciendo uso de la ecuación 1.25, las ecuaciones de movimiento del manipulador virtual son:

$$\begin{aligned}
\tau_1 &= m_{11}\ddot{q}_1 - 2\dot{q}_1\dot{q}_2(m_3c_2s_2L_1^2 + m_2c_2s_2x_2^2 + m_3c_{23}s_{23}x_3^2 + m_3s_{223}L_1x_3) \\
&\quad - 2\dot{q}_1\dot{q}_3(m_3c_{23}s_{23}x_3^2 + m_3c_2s_{23}L_1x_3) \\
\tau_2 &= m_{22}\ddot{q}_2 + m_{23}\ddot{q}_3 + \dot{q}_1^2(m_3c_2s_2L_1^2 + m_2c_2s_2x_2^2 + m_3c_{23}s_{23}x_3^2 + m_3s_{223}L_1x_3) \\
&\quad - m_3s_3L_1x_3\dot{q}_3^2 - 2m_3s_3L_1x_3\dot{q}_2\dot{q}_3 + m_3g(c_2L_1 + c_{23}x_3) + m_2gc_2x_2 \\
\tau_3 &= m_{23}\ddot{q}_2 + m_{33}\ddot{q}_3 + \dot{q}_1^2(m_3c_{23}s_{23}x_3^2 + m_3c_2s_{23}L_1x_3) + m_3gc_{23}x_3 \\
&\quad + m_3s_3L_1x_3\dot{q}_2^2
\end{aligned} \tag{1.38}$$

Agrupando los términos, las ecuaciones de movimiento pueden ser escritas como:

$$\begin{aligned}
\begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ b_{21} & 0 & b_{23} \\ b_{31} & -b_{23} & 0 \end{bmatrix} \begin{bmatrix} \dot{q}_1^2 \\ \dot{q}_2^2 \\ \dot{q}_3^2 \end{bmatrix} + \begin{bmatrix} -2b_{21} & -2b_{31} & 0 \\ 0 & 0 & 2b_{23} \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{q}_1\dot{q}_2 \\ \dot{q}_1\dot{q}_3 \\ \dot{q}_2\dot{q}_3 \end{bmatrix} + \begin{bmatrix} 0 \\ g_2 \\ g_3 \end{bmatrix} \\
= \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix}
\end{aligned}$$

$$b_{21} = m_3c_2s_2L_1^2 + m_2c_2s_2x_2^2 + m_3c_{23}s_{23}x_3^2 + m_3s_{223}L_1x_3 \tag{1.39}$$

$$b_{23} = -m_3s_3L_1x_3$$

$$b_{31} = m_3c_{23}s_{23}x_3^2 + m_3c_2s_{23}L_1x_3$$

$$g_2 = m_3g(c_2L_1 + c_{23}x_3) + m_2gc_2x_2$$

$$g_3 = m_3gc_{23}x_3$$

Esta agrupación permite la generación de lo que se conoce como la ecuación general de la dinámica del manipulador en forma matricial, resumiéndose en:

$$\begin{aligned}
\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q})\dot{\mathbf{q}}^2 + \mathbf{B}(\mathbf{q})[\dot{\mathbf{q}}\dot{\mathbf{q}}] + \mathbf{G}(\mathbf{q}) &= \boldsymbol{\tau} \\
\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{N}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{G}(\mathbf{q}) &= \boldsymbol{\tau}
\end{aligned} \tag{1.40}$$

De manera que:

- ✓ **M(q)**: matriz de masas, simétrica definida positiva, que implica la existencia de la matriz inversa. Cada término m_{ii} de su diagonal representa la inercia vista desde el eslabón i ; los términos fuera de la diagonal representan los acoplamientos entre eslabones.
- ✓ **τ**: es el vector de pares de las articulaciones, el cual está definido por $\boldsymbol{\tau} = [\tau_1 \ \tau_2 \ \tau_3]^T$.
- ✓ **C(q)**: es la matriz que contiene las fuerzas centrífugas inmersas en los movimientos de los eslabones del robot.
- ✓ **B(q)**: es la matriz que contiene las fuerzas de Coriolis involucradas en los movimientos de los eslabones del robot.
- ✓ **G(q)**: es un vector 3x1 el cual contiene los pares o fuerzas gravitacionales, siendo estos significativos en robots que se mueven en el plano vertical.
- ✓ **N(q, q̇)**: vector que agrupa los efectos de las fuerzas centrífugas y de Coriolis.

Para facilitar la simulación del comportamiento dinámico del robot, a partir de la ecuación 1.39, se obtienen las ecuaciones dinámicas en el espacio de estados [11].

$$\begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} = M^{-1} \left\{ \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix} - \begin{bmatrix} 0 & 0 & 0 \\ b_{21} & 0 & b_{23} \\ b_{31} & -b_{23} & 0 \end{bmatrix} \begin{bmatrix} \dot{q}_1^2 \\ \dot{q}_2^2 \\ \dot{q}_3^2 \end{bmatrix} - \begin{bmatrix} -2b_{21} & -2b_{31} & 0 \\ 0 & 0 & 2b_{23} \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{q}_1\dot{q}_2 \\ \dot{q}_1\dot{q}_3 \\ \dot{q}_2\dot{q}_3 \end{bmatrix} - \begin{bmatrix} 0 \\ g_2 \\ g_3 \end{bmatrix} \right\}$$

$$\ddot{q}_1 = \frac{\tau_1 + 2b_{21}\dot{q}_1\dot{q}_2 + 2b_{31}\dot{q}_1\dot{q}_3}{m_{11}} \quad (1.41)$$

$$\ddot{q}_2 = \frac{[m_{23}(b_{31}\dot{q}_1^2 - b_{23}\dot{q}_2^2 + g_3 - \tau_3)] - [m_{33}(b_{21}\dot{q}_1^2 + b_{23}\dot{q}_3^2 + 2b_{23}\dot{q}_2\dot{q}_3 + g_2 - \tau_2)]}{-m_{23}^2 + m_{22}m_{33}}$$

$$\ddot{q}_3 = \frac{[m_{23}(b_{21}\dot{q}_1^2 + b_{23}\dot{q}_3^2 + 2b_{23}\dot{q}_2\dot{q}_3 + g_2 - \tau_2)] - [m_{22}(b_{31}\dot{q}_1^2 - b_{23}\dot{q}_2^2 + g_3 - \tau_3)]}{-m_{23}^2 + m_{22}m_{33}}$$

La Ecuación 1.40 de la dinámica del sistema es una forma de representación a la cual se le puede incluir los efectos como: la fricción en las articulaciones y los efectos de la dinámica de los actuadores. En la bibliografía puede haber otras representaciones, como en [12] en la cual la dinámica del manipulador puede ser escrita en función de los parámetros del sistema (Lewis [8]), siendo una representación para proceso de determinación de parámetros, o para implementación de controladores adaptativos como se verá en el Capítulo 2.

$$\boldsymbol{\tau} = \mathbf{W}(q, \dot{q}, \ddot{q})\boldsymbol{\varphi} = \mathbf{M}\ddot{\mathbf{q}} + \mathbf{N}_m\dot{\mathbf{q}} + \mathbf{G} \quad (1.42)$$

Donde la matriz \mathbf{W} contiene términos temporales, y $\boldsymbol{\varphi}$ es el vector de parámetros del robot. El término \mathbf{N}_m , presenta una factorización de la matriz \mathbf{N} , la cual según [8] puede ser determinada por:

$$\mathbf{N}_m = \frac{1}{2}[\dot{\mathbf{M}} + \mathbf{S}] \quad (1.43)$$

Donde \mathbf{S} satisface la propiedad antisimétrica, y se determina por medio de:

$$\mathbf{S} = \mathbf{U} - \mathbf{U}^T; \quad \mathbf{U} \in \mathbb{R}^{3 \times 3} \quad (1.44)$$

$$\mathbf{U} = (\mathbf{I}_{3 \times 3} \otimes \mathbf{q}^T) \left(\frac{\partial \mathbf{M}}{\partial \mathbf{q}} \right) \quad (1.45)$$

\otimes es el producto Kronecker.

Finalmente, realizando operaciones con lo obtenido en la Ecuación 1.39, la matriz \mathbf{N}_m es:

$$\mathbf{N}_m = \begin{bmatrix} v_{11} & v_{12} & v_{13} \\ v_{21} & v_{22} & v_{23} \\ v_{31} & v_{32} & v_{33} \end{bmatrix}$$

$$\begin{aligned} v_{11} &= -\dot{q}_3 b_{31} - \dot{q}_2 b_{21} & v_{21} &= \dot{q}_1 b_{21} & v_{31} &= \dot{q}_1 b_{31} \\ v_{12} &= -\dot{q}_1 b_{21} & v_{22} &= \dot{q}_3 b_{23} & v_{32} &= -\dot{q}_2 b_{23} \\ v_{13} &= -\dot{q}_1 b_{31} & v_{23} &= \dot{q}_2 b_{23} + \frac{1}{2} \dot{q}_3 b_{23} & v_{33} &= 0 \end{aligned} \quad (1.46)$$

1.8. Control de manipuladores

Un área muy estudiada en el ámbito de la robótica es el control de manipuladores, ya que estos son sistemas MIMO cuya dinámica es no lineal. En función de las aplicaciones, sobre todo en el ámbito industrial, existe un gran interés por el diseño de sistemas de control, ya que los manipuladores en general desarrollan tareas que requieren alto desempeño con buena exactitud [3, 4, 7, 8].

En general se puede hablar de dos problemas de control en manipuladores: control de posición y control de trayectoria

1.8.1. Control de posición

En este caso existe una posición articular deseada (set point) \mathbf{q}_d constante en el tiempo, por lo que el controlador debe posicionar el extremo final del robot en dicha posición desde cualquier posición inicial, de manera que permanezca constante indefinidamente. Este control puede extenderse al seguimiento de puntos consecutivos de manera que el robot siga una trayectoria, conocido también como un control punto a punto.

En este contexto, se requiere que la aplicación del controlador genere un punto de equilibrio asintóticamente estable. Cuya ley de control proporcione los pares adecuados que se apliquen a las articulaciones, para que la posición actual $\mathbf{q}(t)$ y la velocidad articular $\dot{\mathbf{q}}(t)$ tiendan asintóticamente a la posición deseada \mathbf{q}_d con velocidad cero.

$$\lim_{t \rightarrow \infty} \begin{bmatrix} \mathbf{q}(t) \\ \dot{\mathbf{q}}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{q}_d \\ 0 \end{bmatrix} \quad (1.47)$$

1.8.2. Control de trayectoria

También llamado control de movimiento, cuyo objetivo es determinar los pares de manera que la posición articular $\mathbf{q}(t)$ siga una posición $\mathbf{q}_d(t)$ variable en el tiempo. Es decir, se busca que el manipulador se mueva en su espacio de trabajo siguiendo una trayectoria establecida, de manera que las posiciones y velocidades de las articulaciones sigan con exactitud a las posiciones y velocidades deseadas:

$$\lim_{t \rightarrow \infty} \begin{bmatrix} \tilde{\mathbf{q}} \\ \dot{\tilde{\mathbf{q}}} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (1.48)$$

Donde, $\tilde{\mathbf{q}}$ representa el error de posición, es decir la diferencia entre la posición deseada $\mathbf{q}_d(t)$ y la posición actual $\mathbf{q}(t)$, mientras que $\dot{\tilde{\mathbf{q}}}$ es el error de velocidad angular. Siendo estos dos parámetros los que involucran el control de movimiento. En algunos casos puede especificarse como parte del controlador un lazo para control de aceleración, apareciendo un parámetro del error de aceleración $\ddot{\tilde{\mathbf{q}}}$.

Para el problema de control de manipuladores ya sea de posición o trayectoria, existe una gran variedad de esquemas de control. En la Tabla 1.7 se resumen algunos que han sido revisados en la bibliografía y que tienen gran importancia en este campo.

Tabla 1.7. Controladores para manipuladores.

Control de posición	Control PID con desacoplado de articulaciones
	Control PD con compensación de gravedad
	Familia de controladores por moldeo de energía
Control de trayectoria	Familia de controladores por moldeo de energía
	Controlador basado en el modelo: Par calculado
	Controlador adaptativo: Par conmutado adaptativo
	Controlador adaptativo: Inercia Adaptativa

(Fuente: Propia)

El control clásico o lineal generalmente se basa en un modelo linealizado, y proporciona buenos resultados en ciertos rangos de operación, siendo muy utilizado por su simplicidad. Por ejemplo, se tiene el control por desacoplo articular, el cual implementa un lazo de control (comúnmente PID) por cada articulación, de manera que los pares de acoplamiento entre articulaciones son tratados como perturbaciones. La desventaja de estos esquemas es que presentan buena respuesta solo en pequeños rangos de operación, mientras que en otros puede mostrar un bajo desempeño e incluso inestabilidad.

En ciertas aplicaciones donde se requiere trabajar en un espacio amplio de trabajo, el control clásico no presenta un buen rendimiento, esto da paso a la aplicación de técnicas de control no lineales que presentan mayor robustez, siendo un campo de interés en este trabajo.

En base al desarrollo computacional, se han logrado varios avances en técnicas de control no lineal, siendo muy útil en problemas actuales en donde las no linealidades empiezan a ser un problema. Estas técnicas permiten incrementar su robustez ante perturbaciones, así como cuando existe incertidumbre de parámetros del sistema.

Existen varios métodos para el diseño de sistemas de control no lineal, pues los investigadores continúan desarrollando esquemas que permiten dar solución a las problemáticas antes mencionadas, proporcionando ventajas como:

- Ayudar al control de manipuladores que involucran incertidumbres en los parámetros del modelo, debido a variaciones en el tiempo por desgaste, o modificaciones significativas en el sistema.

- Ampliar el rango de operación de control.
- Facilidad en el diseño e implementación gracias a la disponibilidad de computadores de grandes capacidades

En el campo del control no lineal se han tenido grandes desarrollos, pues se tienen varias técnicas como: linealización por realimentación, control adaptativo, modos deslizantes (SMC), entre otras.

En este proyecto se ha enfocado como primer punto la implementación de un controlador basado en el modelo “Par Calculado” (CPC) el cual considera las no linealidades del sistema sin incertidumbre en los parámetros. Con la finalidad de exponer un caso más real en donde haya incertidumbre en los parámetros o variación de la dinámica del sistema, lo cual puede suceder por la sujeción de un objeto por parte del manipulador, se ha considerado la implementación de un controlador adaptativo “Inercia Adaptativa” CIA.

Ambos controladores presentan un sustento teórico y práctico para su implementación en un manipulador, pues no requieren realimentación del vector de aceleraciones articulares evitando el ingreso de ruido en sistemas implementado en computadores.

1.8.3. Control por par calculado – CPC

Este algoritmo de control es un método de linealización por realimentación, el cual emplea un término de control no lineal para “cancelar” una no linealidad del sistema, buscando que el sistema en bucle cerrado se reduzca a un sistema idealmente lineal.

En estos esquemas se consideran los acoplamientos entre articulaciones y buscan eliminar las no linealidades causadas por los efectos centrífugos, Coriolis y gravitacionales, proporcionando un par adecuado basado en las mismas no linealidades. Es por esa razón que estos controladores emplean el modelo dinámico del manipulador.

1.8.4. Control de inercia adaptativa – CIA

Los controladores adaptativos presentan mayor robustez, pues estos se adaptan a los cambios que existen en la dinámica del sistema. En este proyecto es de interés el caso en el que un manipulador sujete un objeto, algo muy frecuente. Esto altera la dinámica del sistema por la variación de la masa del último eslabón, si se considera el controlador por par calculado, el mismo presentará una disminución en su rendimiento.

Es así como se ha visto usar un esquema de Inercia Adaptativa, el cual estima las masas de los eslabones adaptándose a los cambios existentes por incertidumbre en los parámetros o en el caso de una perturbación como la mencionada anteriormente. Este

esquema estima las masas en función de los errores existentes en el seguimiento de trayectoria. El método de diseño tomado es el de Slotine y Li [8], el cual se basa en la estabilidad vista desde Lyapunov.

Los esquemas de control mencionados se acoplarán al manipulador virtual usado en el proyecto, y el análisis y diseño de estos se presentan con más detalle en el Capítulo 2.

De lo mencionado, en este proyecto se emplea brazaletes para el control del movimiento e interacción con el robot y su entorno virtual usando datos inerciales y señales de EMG, es por tal razón que a continuación, se expondrá conceptos que implican el uso y tratamiento de los datos de los brazaletes.

1.9. Miología

Una neurona motora es la encargada de llevar la orden del cerebro al músculo [34] para que un grupo de fibras musculares se contraigan [34]. Una unidad motora es un conjunto de fibras musculares inervadas por una misma neurona motora. En la Figura 1.13 se muestra una gráfica de una unidad motora. Las unidades motoras son responsables de producir la fuerza y el movimiento de los músculos. Un músculo esquelético está formado por varias unidades motoras, las cuales a su vez están formadas por fibras musculares. Cada neurona motora hace sinapsis con numerosas fibras en el interior del músculo. Como existen muchas más fibras que neuronas motoras, los axones motores se ramifican para poder contactar con muchas fibras diferentes y con una amplia distribución. Los axones son el medio por donde viajan los impulsos nerviosos hacia otros tipos de células.

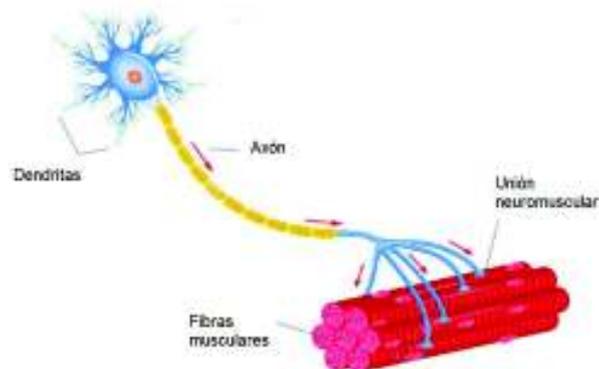


Figura 1.13. Unidad motora.

(Fuente: <https://elaticodejulie.wordpress.com/datos/la-neurona-los-musculos-y-la-unidad-motora/>)

1.9.1. Músculos del antebrazo

Aquí se da una breve descripción de los músculos del antebrazo que están involucrados directamente con el movimiento de la mano, y específicamente al momento de realizar los gestos que se propone utilizar en este proyecto para el control del manipulador virtual.

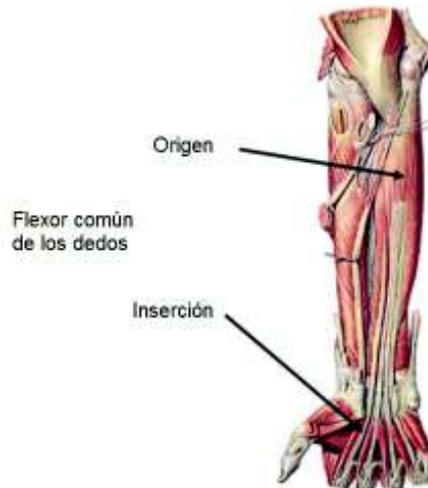


Figura 1.14. Flexor común profundo de los dedos.
 (Fuente: <http://max-flores-gamboa.blogspot.com/2012/04/musculos-del-miembro-superior.html>)

El músculo flexor común (Figura 1.14), se origina en la cara anterior a los $\frac{3}{4}$ proximales del cúbito. Se inserta mediante 4 tendones que se separan al pasar por la muñeca en la base de la tercera falange de los 4 últimos dedos [49].



Figura 1.15. Músculo pronador redondo.
 (Fuente: <http://max-flores-gamboa.blogspot.com/2012/04/musculos-del-miembro-superior.html>)

El músculo pronador redondo (Figura 1.15) es un músculo superficial del antebrazo, localizado en la parte externa de la primera región anterior del antebrazo; aplanado y oblicuo. Cruza en diagonal la cara anterior del antebrazo. Está situado en la parte superior y media del antebrazo [49].

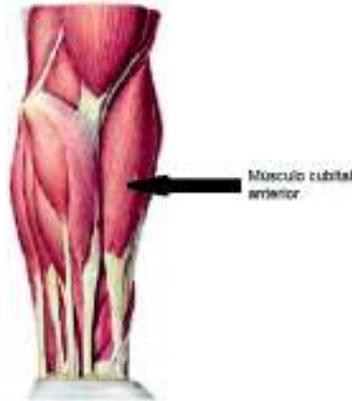


Figura 1.16. Músculo cubital anterior.

(Fuente: <http://max-flores-gamboa.blogspot.com/2012/04/musculos-del-miembro-superior.html>)

El músculo cubital anterior (Figura 1.16) se localiza en el borde interno del antebrazo. Es muy voluminoso y visible por detrás. Sobre la muñeca actúa realizando flexión con aducción [49].



Figura 1.17. Músculo extensor radial corto.

(Fuente: <http://max-flores-gamboa.blogspot.com/2012/04/musculos-del-miembro-superior.html>)

El músculo extensor radial corto del carpo (Figura 1.17) o segundo radial externo es un músculo del antebrazo que extiende y abduce la mano, también imprime un movimiento de supinación del antebrazo. Se ubica bajo el músculo extensor radial largo del carpo y produce la extensión de la muñeca [49].

1.9.2. Electromiografía

La electromiografía es la técnica de registro gráfico de la actividad eléctrica producida por los músculos esqueléticos (Figura 1.18). En la electromiografía se utilizan dispositivos diminutos denominados electrodos para traducir estas señales en gráficos, sonidos o valores numéricos de potencial.

Los potenciales musculares son utilizados en este trabajo para desarrollar una interfaz de comunicación entre el usuario y un manipulador virtual. Interfaces de este tipo permiten a

cualquier usuario controlar sistemas robóticos o mecatrónicos mediante la contracción de determinados músculos cuando se realiza un movimiento.

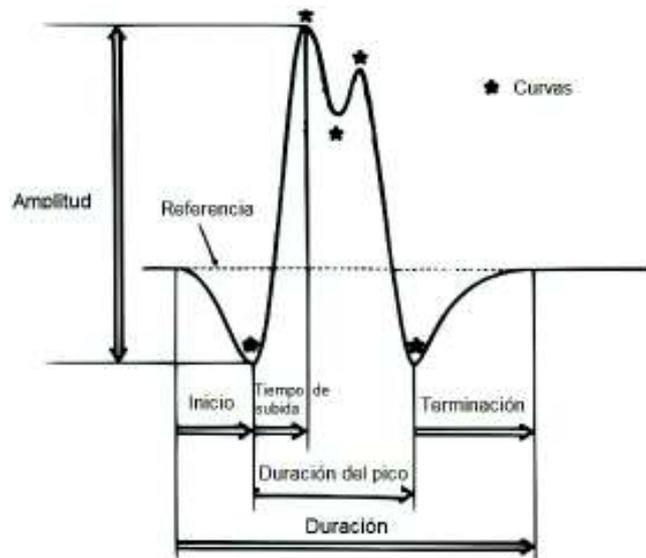


Figura 1.18. Registro de EMG de acción de una unidad motora.

(Fuente: The physiological and technical basis of electromyography. W. F. Brown Boston, Butterworth, 1984)

La Figura 1.18 muestra una onda eléctrica que es transmitida por la membrana celular y que llega hasta el final del axón, provocando la emisión de neurotransmisores a la membrana de la neurona postsináptica, generando un potencial de acción que lleva una orden o información hacia un grupo muscular.

La duración se define como el tiempo desde la deflexión inicial al retorno a la línea de referencia y refleja el número y dispersión espacial de las fibras musculares de la Unidad Motora (UM).

La amplitud se mide entre pico y pico y corresponde a la actividad de un relativo número de fibras musculares.

1.10. MYO Armband

El brazalete de censado MYO Armband es un dispositivo propietario que se usa para la medición de señales EMG de los músculos del antebrazo, y es distribuido en el mercado por la empresa Thalmic Labs Inc [35].

El MYO Armband (véase Figura 1.19) es un brazalete con ocho electrodos tipo seco diferenciales de superficie; además, posee una unidad de medición inercial (IMU) de 9 grados de libertad (acelerómetro, magnetómetro y giroscopio en x, y, z).

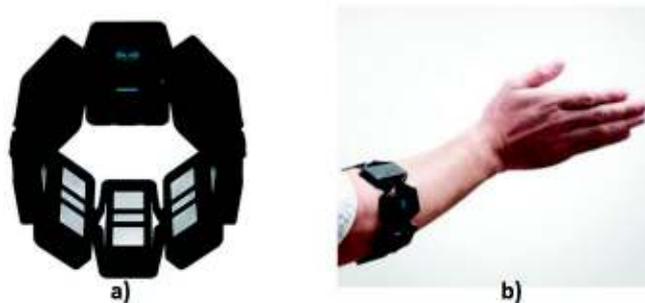


Figura 1.19. a) Brazaletes MYO; b) Brazaletes MYO en el antebrazo de una persona.
(Fuente: Thalmic Labs, 2014)

Las características del brazaletes MYO son:

- 8 electrodos secos diferenciales de superficie.
- IMU (acelerómetro, magnetómetro y giroscopio en x, y, z).
- Frecuencia de muestreo del EMG: 200 Hz.
- Frecuencia de muestreo de la IMU: 50 Hz.
- Comunicación Bluetooth.
- Realimentación háptica a través de vibraciones.

El dispositivo MYO está compuesto por un grupo de sensores de grado médico de acero inoxidable de alta sensibilidad. Estos dispositivos son capaces de censar usando su propio software (MYO Connect) la actividad eléctrica de los músculos EMG y utilizar estas señales como instrucciones para controlar dispositivos mecatrónicos de forma inalámbrica.

Además, también está equipado con un procesador ARM Cortex M4 e indicadores LED. Cuenta con una batería de litio que se carga a través de micro USB y que ofrece una autonomía de un día aproximadamente.

Durante la configuración inicial del software MYO Connect se muestra los diferentes gestos que reconoce el brazaletes MYO: pellizco con los dedos índice y corazón (Figura 1.20a), mano abierta (Figura 1.20b), mano en ángulo de 90° hacia adentro (Figura 1.20c) o hacia afuera (Figura 1.20d) y puño (Figura 1.20e),.

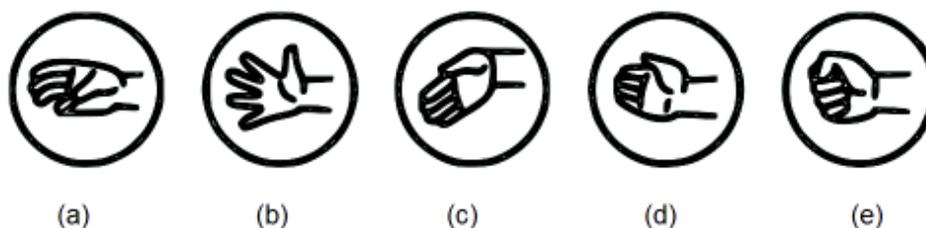


Figura 1.20. Gestos reconocidos por default en el brazaletes MYO.
a) Pitch, b) Spread, c) Wave in, d) Wave out, e) Fist.
(Fuente: Propia)

La necesidad de reconocer nuevos gestos y mejorar la exactitud en la clasificación de los mismos, son motivos para plantear la creación de un nuevo sistema de tratamiento y manejo de los datos del brazalete para este proyecto.

La IMU disponible en el brazalete MYO tiene la combinación de sensores tales como: giroscopio, acelerómetro, y sensores magnéticos de tres ejes (x, y, z). Dentro de los datos de salida proporcionados por el sensor se tiene: matriz de rotación, cuaterniones, datos puros del giroscopio, acelerómetro y magnetómetro. A partir de los cuaterniones proporcionados por el sensor es de interés obtener los ángulos de Euler que proporcionan la orientación de un cuerpo en el espacio.

Para el proyecto se emplea 2 brazaletes, uno que se ubica en el antebrazo y que se encarga de registrar la actividad bioeléctrica del antebrazo, para el procesamiento y reconocimiento de 6 gestos de la mano (MYO 1). El otro brazalete se ubica en el brazo (MYO 2) para obtener la orientación del sensor IMU, que junto con la orientación del sensor IMU del primer brazalete permite controlar los movimientos del robot en el espacio 3D.

1.11. Reconocimiento de gestos

El reconocimiento de gestos es un problema que puede ser resuelto usando aprendizaje computacional [36] y tiene como objetivo interpretar los gestos humanos a través de algoritmos matemáticos [50], los cuales permiten obtener un resultado según el gesto realizado. Los gestos pueden ser cualquier movimiento o estado del cuerpo, pero comúnmente se originan en la cara o en la mano.

En general los sistemas de interacción humano-computador, como los manipuladores, están compuestos por: la señal fisiológica de control (señal EMG para este proyecto), una etapa de procesamiento de la señal y una etapa de control. Las señales fisiológicas son analizadas y convertidas en comandos de control, y el dispositivo de salida realiza una acción de acuerdo al comando recibido. Todas estas etapas están vinculadas por lazos de retroalimentación.

1.12. Etapas de los sistemas de clasificación de gestos

La clasificación de gestos involucra varios procesos previos. Estos procesos son la adquisición de datos propiamente dicha, un preprocesamiento, la extracción de características y la clasificación. En la Figura 1.21 se puede apreciar cada una de las etapas del sistema de clasificación de gestos.

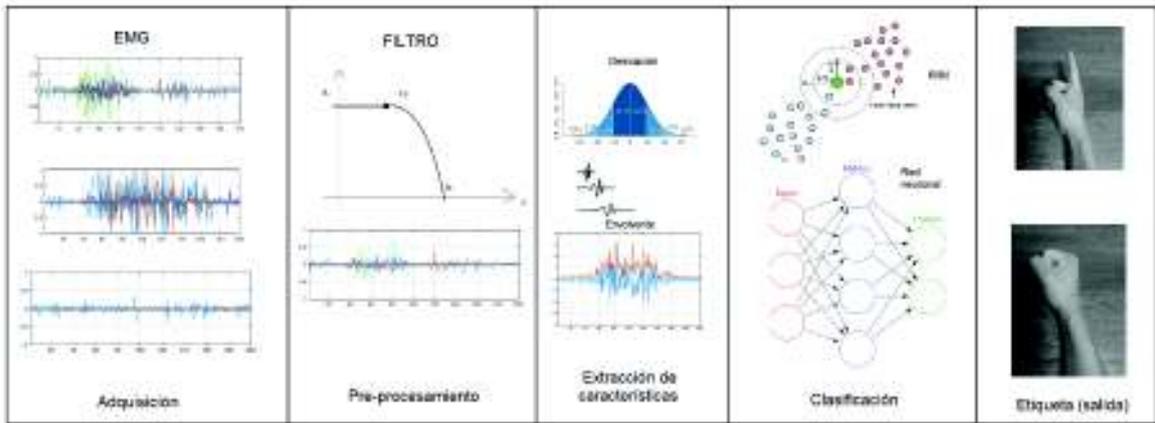


Figura 1.21. Etapas del sistema de clasificación de gestos.
(Fuente: Propia)

1.12.1. Adquisición

La adquisición consiste en la colocación de los electrodos y censado de las señales EMG. Mediante el brazalete MYO se realiza este proceso. El brazalete MYO brinda la posibilidad de tener un sistema compacto y ergonómico. Para el censado existe diferentes configuraciones posibles de electrodos: monopolar y bipolar. El brazalete MYO tiene una configuración bipolar que toma como señales de entrada dos puntos de contacto en el antebrazo de una persona y toma otro punto como señal de referencia. En la Figura 1.22 se puede ver los tipos de montaje monopolar y bipolar. En la Figura 1.23 se puede apreciar el montaje bipolar de los electrodos en el brazalete MYO.

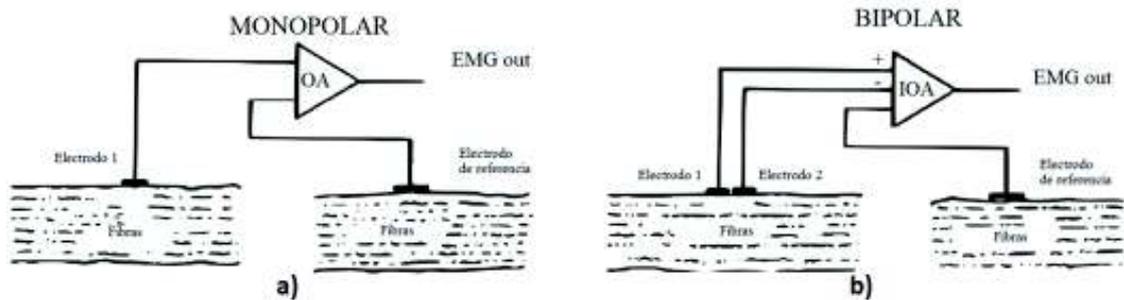


Figura 1.22. Configuraciones de electrodos para EMG: a) mono-polar y b) bi-polar.
(Fuente: Propia)

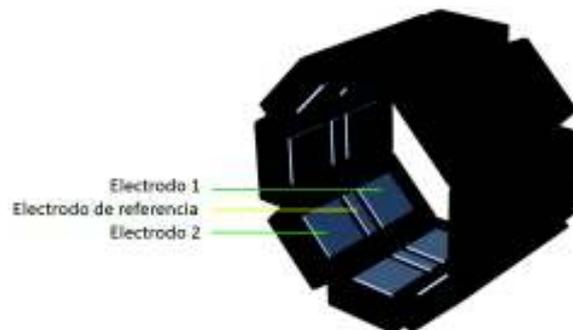


Figura 1.23. Configuración bi-polar de electrodos del brazalete MYO.
(Fuente: Propia)

1.12.2. Pre-procesado y extracción de características

Mediante el uso de análisis estadísticos aplicados a las señales de EMG, se obtiene características de cada uno de los gestos realizados por la persona. Estas características sirven para diferenciar algunas tendencias entre los diferentes gestos. Para un mejor análisis de los datos y antes de obtener las variables estadísticas, es necesario que estos datos sean filtrados usando un filtro pasa bajas. En este caso se usa un filtro de Butterworth digital pasa bajas con frecuencia de corte de 50 Hz. En la Figura 1.24a, se puede observar una señal de EMG pura, mientras que en la Figura 1.24b, se observa la señal EMG a la que se le ha aplicado un filtro pasa bajas de 50Hz.

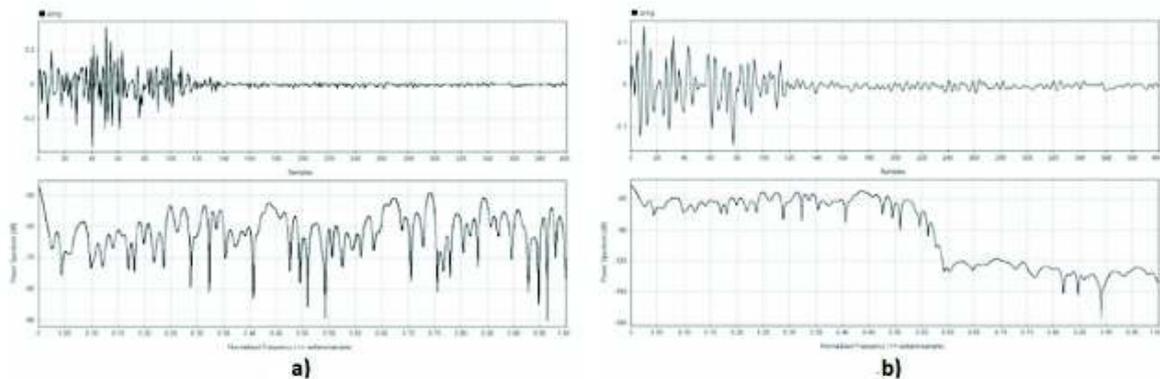


Figura 1.24. a) EMG sin filtrar, b) EMG filtrada $f_c=50\text{Hz}$.
(Fuente: Propia)

La extracción de características es un proceso por el cual los atributos de las señales EMG son calculados y recolectados en un formato de vector compacto. El objetivo de la extracción de características es crear una representación con información útil, manejable y significativa de la señal EMG original sin pérdida de información relevante, con el objeto de incrementar el desempeño de la fase de clasificación [37]. El éxito de la fase de extracción de características depende básicamente de la calidad de las señales medidas por eso la importancia del censado y pre procesamiento correcto de las señales EMG.

Un proceso estacionario es un proceso estocástico, cuya distribución de probabilidad en un instante de tiempo fijo o una posición fija es la misma para todos los instantes de tiempo o posiciones. En consecuencia, parámetros tales como la media y la varianza no varían a lo largo del tiempo.

La no estacionariedad de la señal EMG hace difícil la extracción de parámetros característicos que representen los eventos que se desea monitorear. Hasta el momento no se ha podido obtener estrictamente un grupo de parámetros y un análisis que refleje con gran acierto las características de la señal EMG medida. Debido a esto es necesario elegir un conjunto de características para ingresarlas a una etapa de clasificación de

patrones [38]. Para la extracción de características se usa análisis como la media, la desviación estándar, la envolvente de la señal, la energía, la densidad de potencia, el valor medio absoluto.

1.12.3. Clasificación

La fase de clasificación es un componente importante de los sistemas inteligentes. Una vez definido el experimento, recogidas y pre procesadas las señales usando diferentes técnicas, y definido el vector de características más representativo, el último paso es la implementación de un clasificador que pueda decidir a qué clase corresponde cada una de las señales o gestos realizados. De este modo se genera una nueva señal para el control de dispositivos de salida (prótesis, brazos robóticos).

La entrada al clasificador es el conjunto de características (patrón de entrada) extraídas en el paso anterior (extracción de características) y la salida es una etiqueta de la pertenencia de un objeto, gesto o señal a una de las categorías a clasificar en este proyecto.

1.12.3.1. Clasificador

El clasificador, también llamado discriminador de clases, es el encargado de asignar clases (etiquetas) a los patrones de entrada. Para conseguir este propósito se divide el espacio de entrada en regiones de decisión. Idealmente todos los patrones de una misma clase se agruparán en la misma zona del espacio de medida sin que exista una superposición entre las distintas agrupaciones formadas por otras clases.

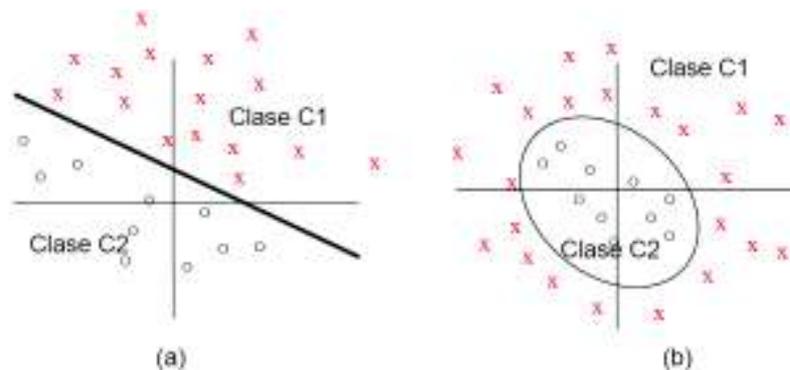


Figura 1.25. a) Sistema separable linealmente, b) Sistema no separable linealmente.
(Fuente: Propia)

En la Figura 1.25a se puede ver dos clases separables con facilidad, mientras que en la Figura 1.25b el proceso de separación de clases es más complejo. Actualmente existe un gran número de técnicas o algoritmos para el reconocimiento de patrones. Entre las técnicas más comúnmente utilizadas están las técnicas tradicionales como los modelos ocultos de Markov, los árboles de decisión, los algoritmos de las k-medias, los métodos

basados en distancias como la distancia Euclídea y distancia Mahalanobis, los métodos estadísticos, de los cuales hacen parte los clasificadores bayesianos simples (naiveBayes), los discriminadores lineales, análisis de componentes principales, análisis factorial.

Entre las técnicas relativamente más recientes están las basadas en técnicas inteligentes como las redes neuronales (como el perceptrón multicapa, la red neuronal probabilística, redes ART, entre otras), máquinas de vectores de soporte (SVM) [26], sistemas Fuzzy, y las técnicas híbridas en las que es posible combinar cualquiera de los métodos ya mencionados, como por ejemplo las redes Neuro – Fuzzy [33].

1.12.3.2. Formas de clasificación (aprendizaje supervisado y no supervisado)

Los métodos no supervisados se refieren a situaciones donde el objetivo es: a partir de una serie de observaciones, establecer la existencia de clases o grupos en los datos (clusters) [39]. En este caso no existe un conocimiento a priori de qué clases existen, ni se dispone de un conjunto de objetos u observaciones etiquetadas, por lo que no se podrán utilizar para diseñar las funciones discriminantes. En el caso de la clasificación supervisada, las clases están predefinidas y se dispone de patrones previamente clasificados. El objetivo consiste en establecer una regla a través de un proceso de entrenamiento, para ubicar nuevas observaciones en alguna de las clases existentes [39].

2. METODOLOGÍA

En este capítulo se expondrá las fases que fueron realizadas en este proyecto con el fin de cumplir con el objetivo principal para la operación remota y control adaptativo de un manipulador virtual de 3 grados de libertad usando brazaletes MYO y reconocimiento de gestos de la mano.

2.1. Adquisición de datos con el brazalete MYO Armband

Para la adquisición de las señales usando el brazalete MYO se debe instalar su SDK (Software Development Kit). Para este proyecto se ha utilizado Windows 8.1 y Matlab versión 2018a. En Matlab se puede utilizar la librería MyoMex, que sirve para probar el funcionamiento del brazalete y asegurar que el sistema está configurado correctamente. En la Figura 2.1 se ve una captura del proceso de instalación del software Myo Connect.



Figura 2.1. Software Myo Connect durante la instalación.
(Fuente: Propia)

El software propietario del brazalete MYO reconoce por default 5 gestos que sirven para controlar diferentes gadgets que se presentan con el software MYO Connect (véase Figura 2.2).

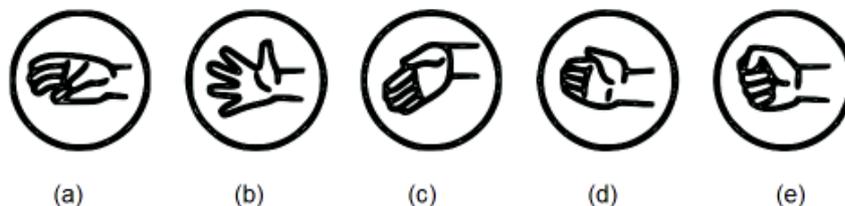


Figura 2.2. Gestos predefinidos que reconoce el brazalete MYO.
(Fuente: Propia)

- a) **Pinch:** el usuario realiza dos golpes rápidos entre el dedo pulgar y medio.
- b) **Spread:** el usuario abre la mano completamente tratando de extender los dedos.
- c) **Wave In:** el usuario gira la mano 90° hacia adentro con los dedos apretados y alargados.

- d) **Wave Out:** el usuario gira la mano 90° hacia afuera con los dedos apretados y alargados.
- e) **Fist:** el usuario cierra completamente su mano, realizando un puño con un poco de fuerza.

2.2. Selección de datos de sensores IMU

Como referencia al trabajo realizado en [10], se usará los ángulos Yaw, Pitch y Roll, para el control de los movimientos del robot. En este caso se ha considerado el uso de los datos proporcionados por el sensor IMU de los dos brazaletes MYO.

- **Brazaletes del antebrazo (MYO 1):** IMU1.
- **Brazaletes del brazo (MYO 2):** IMU2.

De los brazaletes se puede extraer los datos de orientación en forma de una matriz de rotación, o en términos de un cuaternión. En el primer caso existe un problema de singularidad (bloqueo de cardán [10]) cuando el ángulo pitch, está en +/-90°, de manera que en este proyecto se ha hecho uso de cuaterniones evitando el problema de singularidad. Los cuaterniones proporcionan una notación matemática para representar las orientaciones y las rotaciones de objetos en tres dimensiones.

Un cuaternión ψ es una representación de la orientación por medio de 4 parámetros, que permite pasar de un sistema a otro por medio de una rotación sobre un vector definido sobre un sistema de referencia fijo [6].

$$\Psi = q_w + q_x i + q_y j + q_z k \quad (2.1)$$

Donde q_w, q_x, q_y y q_z , son los términos del cuaternión, mientras que i, j y k son los vectores unitarios. Con el fin de obtener los ángulos de orientación requeridos a partir del cuaternión, se ha usado las siguientes expresiones:

$$\begin{aligned} Roll &= \text{atan2}(2(q_w * q_x + q_y * q_z), 1 - 2(q_x^2 + q_y^2)) \\ Pitch &= \sin^{-1}(\max(-1, \min(1, 2(q_w * q_y - q_z * q_x)))) \\ Yaw &= \text{atan2}(2(q_w * q_z + q_x * q_y), 1 - 2(q_y^2 + q_z^2)) \end{aligned} \quad (2.2)$$

Donde la función Atan2 , representa el inverso de la función trigonométrica tangente. Así mismo, para diferenciar los ángulos Roll (R), Pitch (P) y Yaw (Y), de cada IMU, se ha establecido la siguiente notación:

- **IMU1:** P_1, Y_1, R_1 .
- **IMU2:** P_2, Y_2, R_2 .

2.2.1. Asignación de ángulos para control de movimiento

Para la asignación de movimientos y la calibración de las referencias de las IMU, se ha considerado una postura inicial, como se ve en la Figura 2.3, además, se muestra la ubicación de los brazaletes en el antebrazo y brazo.

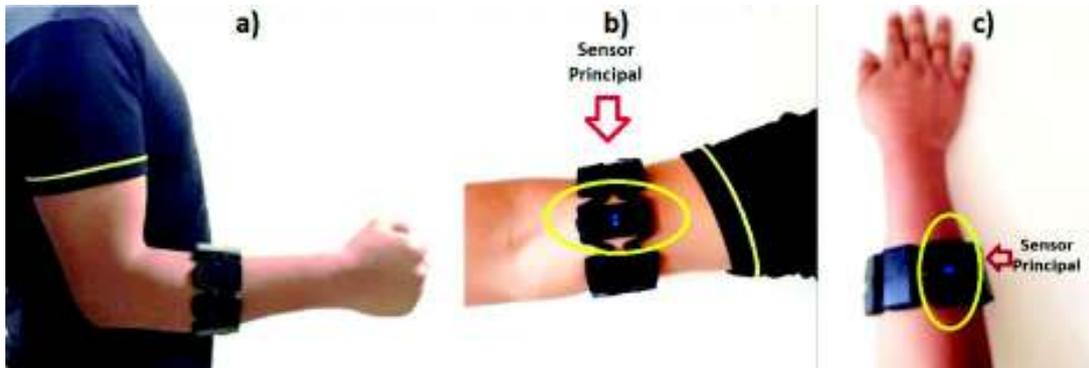


Figura 2.3. a) Postura inicial; b) ubicación del MYO en el brazo; c) ubicación del MYO en el antebrazo.
(Fuente: Propia)

Para el control del ángulo de rotación de la base del robot (q_1), en un rango de -90° a 90° (Tabla 1.4), se ha considerado el ángulo Y_1 extraído del brazaletе 1, Figura 2.4a.

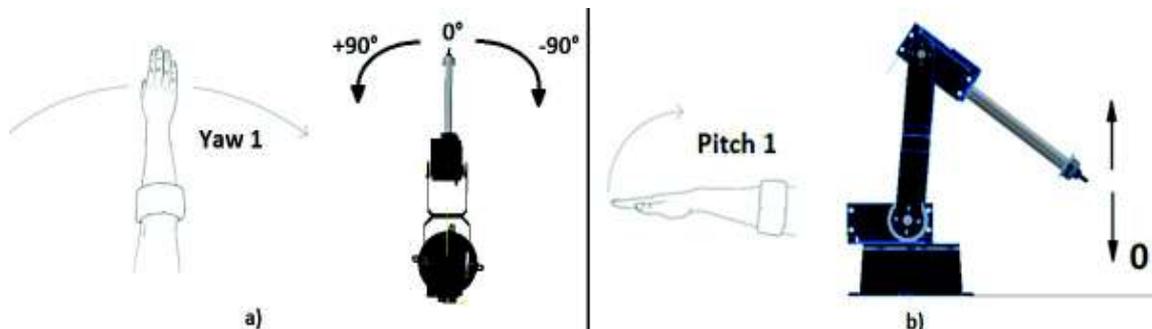


Figura 2.4. Asignación de ángulos: a) Y_1 ; b) P_1 .
(Fuente: Propia)

En base al brazaletе 1 se usa el ángulo pitch (P_1) para el movimiento de subir o bajar el extremo del robot, Figura 2.4b. El rango de operación es desde el nivel horizontal 0, hasta su límite superior que depende del espacio de trabajo diestro (véase Figura 1.11).

Con el uso del ángulo pitch del segundo brazaletе (P_2) ubicado en brazo, se podrá controlar el desplazamiento del extremo del robot sobre la línea horizontal (Figura 2.5) en un rango de 0.1 a 0.3 m definido de acuerdo al espacio de trabajo diestro (Figura 1.11).

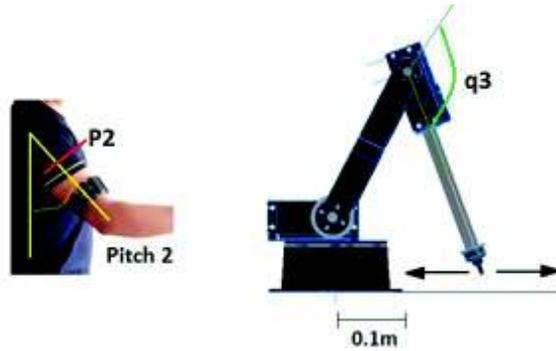


Figura 2.5. Asignación del ángulo P_2 .
(Fuente: Propia)

P_1 y P_2 están ligados directamente a los movimientos realizados por el usuario, con el fin de facilitar la realización de los mismos y que sean cómodos para el usuario, se ha considerado los siguientes rangos angulares:

- P_1 : 0-60°.
- P_2 : 0-30°

Además, como los ángulos P_1 y P_2 no están ligados directamente a las articulaciones del robot, se genera un problema de cinemática inversa, el cual se expone a continuación.

2.3. Cinemática inversa

Con los ángulos P_1 y P_2 definidos, se pretende determinar el valor angular de las articulaciones q_2 y q_3 , las cuales generan un plano 2D como se ve en la Figura 2.6. De esta manera, se requiere de dos componentes para establecer la ubicación del extremo del robot (A_x y A_z) referenciados al sistema de referencia ubicado en la articulación 1.

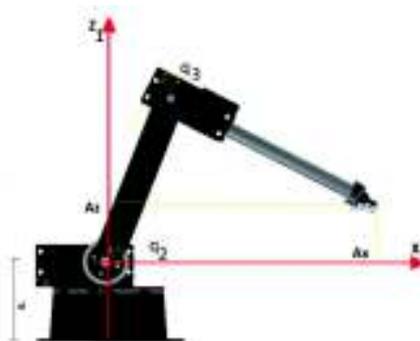


Figura 2.6. Plano generado por P_1 y P_2 .
(Fuente: Propia)

Además, con referencia a la Sección 2.2, mediante el uso de P_1 y P_2 , se busca generar las componentes A_x y A_z usando relaciones geométricas. De manera que de acuerdo a la ubicación de los brazaletes y asignación de ángulos (Sección 2.2.1), con cada ángulo, se

forma un triángulo rectángulo (Figura 2.7), lo cual permitirá mapear un punto en el espacio de trabajo del robot y a su vez permitirá determinar las componentes A_x y A_z .

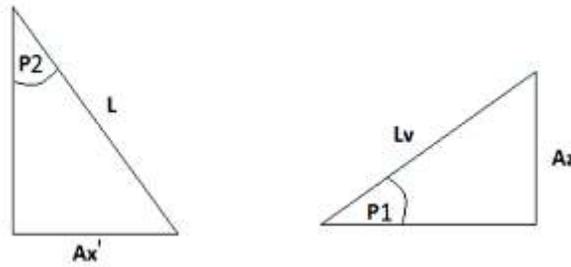


Figura 2.7. Triángulo con Pitch 1; Triángulo con Pitch 2.
(Fuente: Propia)

Dónde, L y L_v son las hipotenusas de los triángulos de la Figura 2.7 y son valores numéricos que se consideran para los cálculos posteriores.

En base a la Figura 1.11, el robot, no puede tener una coordenada $A_x=0$, por lo que para resolver el triángulo de P_2 , se usa una variable auxiliar A_x' . De esta forma, L se ha tomado como un valor fijo de 0.4m, de manera que cuando P_2 se máximo, es decir 30° , A_x no supere el límite del espacio de trabajo diestro (0.3m), por lo tanto:

$$A_x = 0.1 + 0.4\text{sen}(P_2) \quad (2.3)$$

Debido a las limitaciones del espacio de trabajo (Figura 1.11), L_v está delimitada, de manera que:

$$L_v = \sqrt{0.3^2 - A_x^2} \quad (2.4)$$

Por lo tanto, la coordenada A_z se determina mediante:

$$A_z = L_v\text{sen}(P_1) \quad (2.5)$$

Como las coordenadas A_x y A_z especifican un punto en el espacio de trabajo del robot, en base a este, el robot deberá posicionarse en dicho punto calculando numéricamente q_2 y q_3 . Dado que el sistema de referencia 1 está desplazado la distancia d_1 del sistema de referencia 0, se considera que $A_z' = A_z - d_1$, siendo útil para la simulación en Matlab.

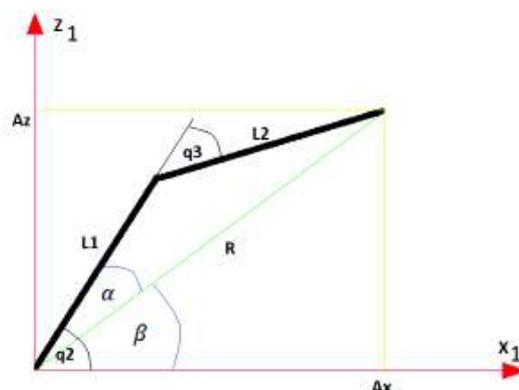


Figura 2.8. Plano 2D del robot.
(Fuente: Propia)

En la Figura 2.8 se muestra el esquema geométrico conformado por los eslabones del robot, los cuales generan triángulos y ángulos, que se usan para determinar q_2 y q_3 en base a lo Ax y Az . De esta manera aplicando relaciones geométricas (ley de cosenos, identidades), q_3 se calcula como:

$$R^2 = Ax^2 + Az^2$$

$$q_3 = -\cos^{-1}\left(\frac{R^2 - L_1^2 - L_2^2}{2L_1L_2}\right) \quad (2.6)$$

La determinación de q_2 implica el conocimiento de α y β de manera que:

$$\beta = \tan^{-1}\left(\frac{Az'}{Ax}\right)$$

$$\alpha = \cos^{-1}\left(\frac{L_2^2 - R^2 - L_1^2}{-2L_1R}\right) \quad (2.7)$$

$$q_2 = \alpha + \beta$$

2.4. Diseño de controladores

2.4.1. Controlador por par calculado (CPC)

Este método emplea el concepto de particionamiento de la ley control [4], en la cual una parte está basada en el modelo que depende de los parámetros del sistema, y con la que se busca que el modelo se simplifique; otra parte basada en un control-servo, siendo muy sencilla de diseñar ya que generalmente es de tipo proporcional derivativo PD (Figura 2.9).

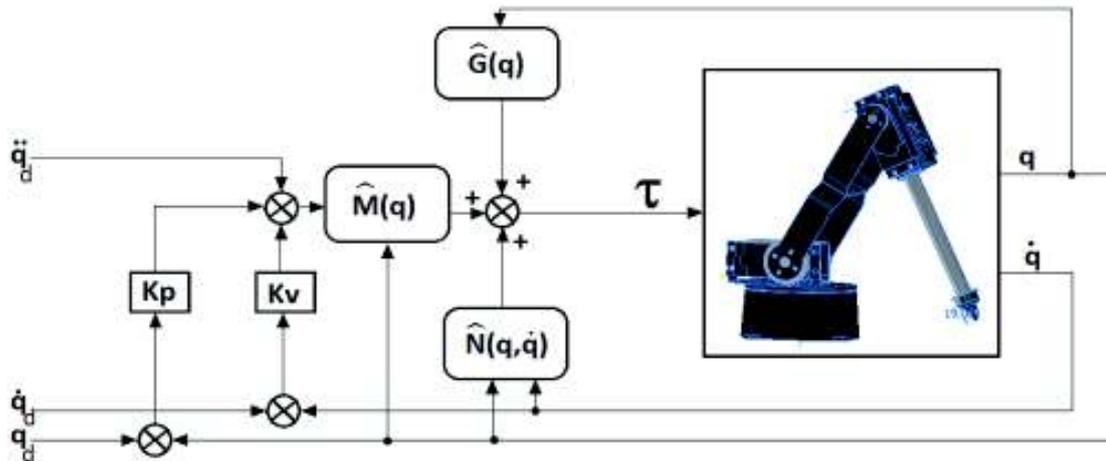


Figura 2.9. Diagrama de bloques del controlador CPC.
(Fuente: Propia)

El controlador PD presenta un punto de equilibrio global y asintóticamente estable. La acción P es proporcional al error de posición, mientras que la acción D tiene un efecto de freno o amortiguamiento a través de la velocidad, es decir a velocidad diferente de cero.

En base a la Ecuación 1.40 del modelo dinámico de un manipulador, se puede establecer una ecuación con parámetros del robot estimados:

$$\hat{M}(q)\ddot{q} + \hat{N}(q, \dot{q}) + \hat{G}(q) = \tau \quad (2.8)$$

Donde $\hat{M}(q)$, $\hat{N}(q, \dot{q})$ y $\hat{G}(q)$, son las matrices y vectores estimados. De manera que, la ley de control del controlador por par calculado es:

$$\tau = \hat{M}(q)\tau_r(e) + \tau_c \quad (2.9)$$

Donde τ_r corresponde a la parte del control del servo, la cual está en función de los vectores de error de velocidad y posición.

$$\begin{aligned} \dot{e} &= \dot{q}_d - \dot{q} \\ e &= q_d - q \end{aligned} \quad (2.10)$$

$$\tau_r = \ddot{q}_d + K_v \dot{e} + K_p e \quad (2.11)$$

K_v y K_p son matrices diagonales 3x3, que contienen las constantes del controlador tipo PD, donde cada termino k_p corresponde a la ganancia proporcional, y k_v corresponde a la ganancia derivativa:

$$\begin{aligned} K_p &= \begin{bmatrix} k_{p1} & 0 & 0 \\ 0 & k_{p2} & 0 \\ 0 & 0 & k_{p3} \end{bmatrix} \\ K_v &= \begin{bmatrix} k_{v1} & 0 & 0 \\ 0 & k_{v2} & 0 \\ 0 & 0 & k_{v3} \end{bmatrix} \end{aligned} \quad (2.12)$$

τ_c es la parte compensadora basada en el modelo, que en este caso contiene términos de fuerzas de Coriolis, centrífuga y gravitacionales.

$$\tau_c = \hat{N}(q, \dot{q}) + \hat{G}(q) \quad (2.13)$$

Por lo tanto, la ley de control en base a las Ecuaciones 2.11 y 2.13 es:

$$\tau = \hat{M}[\ddot{q}_d + K_v \dot{e} + K_p e] + \hat{N} + \hat{G} \quad (2.14)$$

Reemplazando la Ecuación 2.14 en la 1.40, se tiene el sistema en lazo cerrado, dado por:

$$\begin{aligned} \hat{M}[\ddot{q}_d + K_v \dot{e} + K_p e] + \hat{N} + \hat{G} &= M\ddot{q} + N + G \\ [\ddot{q}_d - \ddot{q} + K_v \dot{e} + K_p e] &= \hat{M}^{-1}[(M - \hat{M})\ddot{q} + (N - \hat{N}) + (G - \hat{G})] \end{aligned} \quad (2.15)$$

Si las matrices estimadas se conocen perfectamente, el lado derecho de la anterior ecuación se anula, obteniendo:

$$\ddot{e} + K_v \dot{e} + K_p e = 0 \quad (2.16)$$

De la Ecuación 2.16, se puede concluir que el sistema queda desacoplado, es decir para cada articulación se obtiene un sistema de segundo orden en donde las constantes k_{vi} y k_{pi} puede ser determinadas utilizando metodologías de control clásico [3,4].

Considerando el sistema de segundo orden, el coeficiente de amortiguamiento δ y la frecuencia angular w_n están expresadas por:

$$\delta = \frac{k_v}{2\sqrt{k_p}} \quad (2.17)$$

$$w_n = \sqrt{k_p} \quad (2.18)$$

En los sistemas de segundo orden para lograr un amortiguamiento crítico se tiene que $\delta = 1$, por lo que de la Ecuación 2.17 se obtiene:

$$k_v = 2\sqrt{k_p} \quad (2.19)$$

Además, el tiempo de establecimiento está expresado por:

$$t_s = \frac{\pi}{\delta} w_n \quad (2.20)$$

Finalmente, para lograr un tiempo de respuesta menor a 1 segundo, se considera que $t_s = 0.52$ [s], logrando obtener $k_{pi} = 36$ y $k_{vi} = 12$.

2.4.2. Controlador de inercia adaptativa (CIA)

Uno de los criterios más relevante en la teoría de control es que el sistema permanezca estable. En este contexto se entenderá el mismo si los errores permanecen pequeños o se hacen cero después de transcurrir un intervalo de tiempo.

En la teoría de control para sistemas lineales existen métodos (Bode, Nyquist) para análisis de estabilidad, los cuales no pueden ser aplicados en sistemas no lineales, es por esa razón que en este trabajo se considera el estudio de estabilidad basado en el método de Lyapunov.

2.4.2.1. Estabilidad de Lyapunov: Método Directo

Estudia el comportamiento de sistemas dinámicos, los cuales pueden estar descritos por:

$$\dot{x} = f(t, x) \quad x(0) \in R^n \quad \forall t \geq 0 \quad (2.21)$$

Donde se establece que para cualquier condición inicial $x(0)$ dentro de la zona de atracción, y si el sistema tiene un estado de equilibrio asintóticamente estable, la energía acumulada por el sistema cae con el tiempo hasta alcanzar el punto de equilibrio. Un punto de equilibrio es un punto en el que todas las fuerzas del sistema están en equilibrio.

En base a lo anterior, el método de Lyapunov caracteriza expresiones que se califican como funciones de energía en los alrededores del punto de equilibrio, las cuales

decrementan en las trayectorias del sistema, de manera que la derivada temporal de dichas funciones es negativa hasta alcanzar el equilibrio.

Una función candidata de Lyapunov $V(x)$ para el equilibrio debe cumplir:

- Ser una función definida positiva.
- $\frac{\partial V(x)}{\partial x}$ debe ser continua a x .
- $\frac{\partial V(x)}{\partial t}$ debe existir y ser continua a x .
- $\dot{V}(x)$ debe ser definida negativa.

Con relación al controlador CPC (Ecuación 2.15), las matrices estimadas y reales se cancelan siempre que se conozcan los parámetros del sistema (φ) con gran exactitud. Ahora se analizará un nuevo problema cuando esto no sucede, es decir hay un error en la cancelación entre los parámetros reales y los estimados, $\tilde{\varphi} = \varphi - \hat{\varphi}$.

$$[\ddot{e} + K_v \dot{e} + K_p e] = \hat{M}^{-1} W \tilde{\varphi} \quad (2.22)$$

Donde $\hat{\varphi} = [\hat{m}_1 \hat{m}_2 \hat{m}_3]^T$ es el vector de parámetros estimados, que para este caso se ha considerado las masas de los eslabones. De manera que, considerando $r(t)$ un vector de error de seguimiento de trayectoria (filtered tracking error [8]) dado por:

$$r = \begin{bmatrix} \lambda_1 e_1 + \dot{e}_1 \\ \lambda_2 e_2 + \dot{e}_2 \\ \lambda_3 e_3 + \dot{e}_3 \end{bmatrix} \quad (2.23)$$

Se busca expresar la ecuación dinámica 1.42 en términos de $r(t)$, es decir:

$$r = \Lambda e + \dot{e} \quad (2.24)$$

$$\dot{e} = \dot{q}_d - \dot{q} \quad (2.25)$$

$$\dot{q} = \Lambda e + \dot{q}_d - r \quad (2.26)$$

Donde Λ para este caso, es una matriz diagonal 3x3.

$$\Lambda = \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \quad (2.27)$$

Además, derivando con respecto al tiempo la Ecuación 2.26, se tiene:

$$\ddot{q} = \Lambda \dot{e} + \ddot{q}_d - \dot{r} \quad (2.28)$$

Ahora para el diseño del controlador adaptativo, se usará el análisis de estabilidad, para lo cual la función candidata de Lyapunov debe estar en función del error de seguimiento de trayectoria, así como del error en los parámetros, es decir:

$$V = \frac{1}{2} r^T M r + \frac{1}{2} \tilde{\varphi}^T \Gamma^{-1} \tilde{\varphi}; \quad \Gamma = \begin{bmatrix} \gamma_1 & 0 & 0 \\ 0 & \gamma_2 & 0 \\ 0 & 0 & \gamma_3 \end{bmatrix} \quad (2.29)$$

Donde, en base a la dimensión del vector r , Γ es una matriz diagonal 3x3, con elementos constantes positivos. Derivando la función V con respecto al tiempo, se obtiene:

$$\dot{V} = r^T M \dot{r} + \frac{1}{2} r^T \dot{M} r + \tilde{\varphi}^T \Gamma^{-1} \dot{\tilde{\varphi}} \quad (2.30)$$

Reemplazando las ecuaciones 2.26 y 2.28 en la Ecuación 1.42, y realizando operaciones, se tiene que:

$$M \dot{r} = M(\Lambda \dot{e} + \ddot{q}_d) + N_m(\Lambda e + \dot{q}_d) + G - N_m r - \tau \quad (2.31)$$

De la Ecuación 2.31 se considera que:

$$Y \varphi = M(\Lambda \dot{e} + \ddot{q}_d) + N_m(\Lambda e + \dot{q}_d) + G \quad (2.32)$$

Donde Y es una matriz de funciones temporales como W , y, además, no depende de \ddot{q} .

$$M \dot{r} = Y \varphi - N_m r - \tau \quad (2.33)$$

Substituyendo la Ecuación 2.33 en la Ecuación 2.30, se tiene que:

$$\begin{aligned} \dot{V} &= r^T (Y \varphi - N_m r - \tau) + \frac{1}{2} r^T \dot{M} r + \tilde{\varphi}^T \Gamma^{-1} \dot{\tilde{\varphi}} \\ \dot{V} &= r^T (Y \varphi - \tau) + r^T \left(\frac{1}{2} \dot{M} - N_m \right) r + \tilde{\varphi}^T \Gamma^{-1} \dot{\tilde{\varphi}} \end{aligned} \quad (2.34)$$

Aplicando la propiedad de la matriz antisimétrica y la relación de la Ecuación 1.43 entre N_m y M , se obtiene:

$$\dot{V} = r^T (Y \varphi - \tau) + \tilde{\varphi}^T \Gamma^{-1} \dot{\tilde{\varphi}} \quad (2.35)$$

Con base al análisis de estabilidad [8], se considera un par de control basado en el modelo, dado por:

$$\tau = Y \hat{\varphi} + K_v r \quad (2.36)$$

Reemplazando la Ecuación 2.36 en la Ecuación 2.35:

$$\begin{aligned} \dot{V} &= r^T (Y \varphi - Y \hat{\varphi}) + \tilde{\varphi}^T \Gamma^{-1} \dot{\tilde{\varphi}} - r^T K_v r \\ \dot{V} &= r^T Y \tilde{\varphi} + \tilde{\varphi}^T \Gamma^{-1} \dot{\tilde{\varphi}} - r^T K_v r \\ \dot{V} &= \tilde{\varphi}^T (Y^T r + \Gamma^{-1} \dot{\tilde{\varphi}}) - r^T K_v r \end{aligned} \quad (2.37)$$

Como φ es un vector de parámetros cuyos valores son constantes, su derivada es nula, de manera que:

$$\begin{aligned} \tilde{\varphi} &= \varphi - \hat{\varphi} \\ \dot{\tilde{\varphi}} &= -\dot{\hat{\varphi}} \end{aligned} \quad (2.38)$$

Por lo tanto, para que el sistema sea estable, la ley de adaptación se toma como:

$$\dot{\hat{\varphi}} = \Gamma Y^T r \quad (2.39)$$

De esta forma, se concluye que \dot{V} está definida negativa, demostrando estabilidad en el sistema.

$$\dot{V} = -r^T K_v r \quad (2.40)$$

En base a [8] se demuestra que los errores en la estimación del vector de parámetros están delimitados, con lo que se puede concluir que los errores de seguimiento de trayectoria son asintóticamente estables. $\lim_{n \rightarrow \infty} V = 0$; $\lim_{n \rightarrow \infty} r = 0$; $\lim_{n \rightarrow \infty} e = 0$; $\lim_{n \rightarrow \infty} \dot{e} = 0$. El diagrama de bloques de este controlador se muestra en la Figura 2.10.

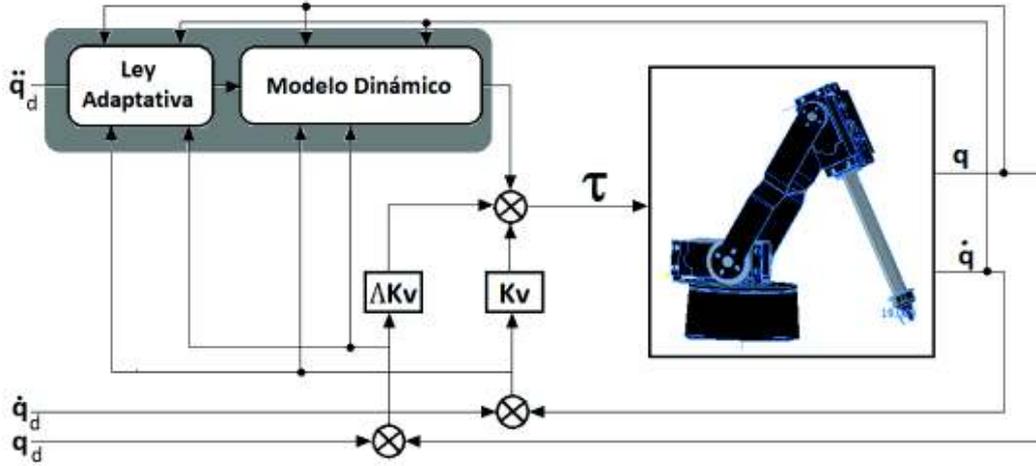


Figura 2.10. Diagrama de bloques del controlador CIA.
(Fuente: Propia)

Para la ley adaptativa de la Ecuación 2.37, se tiene:

$$\Gamma = \begin{bmatrix} \gamma_1 & 0 & 0 \\ 0 & \gamma_2 & 0 \\ 0 & 0 & \gamma_3 \end{bmatrix}; \quad Y = \begin{bmatrix} Y_{11} & Y_{12} & Y_{13} \\ Y_{21} & Y_{22} & Y_{23} \\ Y_{31} & Y_{32} & Y_{33} \end{bmatrix} \quad (2.41)$$

$$\hat{m}_1 = \gamma_1 [Y_{11}(\lambda_1 e_1 + \dot{e}_1) + Y_{21}(\lambda_2 e_2 + \dot{e}_2) + Y_{31}(\lambda_3 e_3 + \dot{e}_3)]$$

$$\hat{m}_2 = \gamma_2 [Y_{12}(\lambda_1 e_1 + \dot{e}_1) + Y_{22}(\lambda_2 e_2 + \dot{e}_2) + Y_{32}(\lambda_3 e_3 + \dot{e}_3)] \quad (2.42)$$

$$\hat{m}_3 = \gamma_3 [Y_{13}(\lambda_1 e_1 + \dot{e}_1) + Y_{23}(\lambda_2 e_2 + \dot{e}_2) + Y_{33}(\lambda_3 e_3 + \dot{e}_3)]$$

Donde:

$$\begin{aligned} Y_{11} &= (x_1^2 + y_1^2)(\lambda_1 \dot{e}_1 + \ddot{q}_{d1}) \\ Y_{12} &= (c_2^2 x_2^2)(\lambda_1 \dot{e}_1 + \ddot{q}_{d1}) - \dot{q}_2 k_1 (\lambda_1 e_1 + \dot{q}_{d1}) - \dot{q}_1 k_1 (\lambda_2 e_2 + \dot{q}_{d2}) \\ Y_{13} &= (c_2^2 L_1^2 + 2c_2 c_{23} L_1 x_3 + c_{23}^2 x_3^2)(\lambda_1 \dot{e}_1 + \ddot{q}_{d1}) - \dot{q}_3 k_3 (\lambda_1 e_1 + \dot{q}_{d1}) - \dot{q}_2 k_2 (\lambda_1 e_1 + \dot{q}_{d1}) \\ &\quad - \dot{q}_1 k_2 (\lambda_2 e_2 + \dot{q}_{d2}) - \dot{q}_1 k_3 (\lambda_3 e_3 + \dot{q}_{d3}) \\ Y_{21} &= Y_{31} = Y_{32} = 0; \\ Y_{22} &= x_2^2 (\lambda_2 \dot{e}_2 + \ddot{q}_{d2}) + k_1 \dot{q}_1 (\lambda_1 e_1 + \dot{q}_{d1}) + c_2 x_2 g \\ Y_{23} &= (L_1^2 + 2k_5 + x_3^2)(\lambda_2 \dot{e}_2 + \ddot{q}_{d2}) + (x_3^2 + k_5)(\lambda_3 \dot{e}_3 + \ddot{q}_{d3}) + \dot{q}_1 k_2 (\lambda_1 e_1 + \dot{q}_{d1}) \\ &\quad - \dot{q}_3 k_4 (\lambda_2 e_2 + \dot{q}_{d2}) - (2k_4 \dot{q}_2 + k_4 \dot{q}_3)(\lambda_3 e_3 + \dot{q}_{d3}) + g(c_2 L_1 + c_{23} x_3) \\ Y_{33} &= (x_3^2 + k_5)(\lambda_2 \dot{e}_2 + \ddot{q}_{d2}) + x_3^2 (\lambda_3 \dot{e}_3 + \ddot{q}_{d3}) + \dot{q}_1 k_3 (\lambda_1 e_1 + \dot{q}_{d1}) + \dot{q}_2 k_4 (\lambda_2 e_2 + \dot{q}_{d2}) + c_{23} g x_3 \end{aligned} \quad (2.43)$$

Definiendo constantes:

$$\begin{aligned} k_1 &= c_2 s_2 x_2^2 \\ k_2 &= (c_2 s_2 L_1^2 + c_{23} s_{23} x_3^2 + s_{223} L_1 x_3) \\ k_3 &= (c_{23} s_{23} x_3^2 + c_2 s_{23} L_1 x_3) \\ k_4 &= (s_3 L_1 x_3) \\ k_5 &= (c_3 L_1 x_3) \end{aligned} \quad (2.44)$$

Por lo tanto, el torque de control es:

$$\begin{aligned}
 \tau_1 &= Y_{11}\hat{m}_1 + Y_{12}\hat{m}_2 + Y_{13}\hat{m}_3 + k_{v1}\lambda_1 e_1 + k_{v1}\dot{e}_1 \\
 \tau_2 &= Y_{21}\hat{m}_1 + Y_{22}\hat{m}_2 + Y_{23}\hat{m}_3 + k_{v2}\lambda_2 e_2 + k_{v2}\dot{e}_2 \\
 \tau_3 &= Y_{31}\hat{m}_1 + Y_{32}\hat{m}_2 + Y_{33}\hat{m}_3 + k_{v3}\lambda_3 e_3 + k_{v3}\dot{e}_3
 \end{aligned}
 \tag{2.45}$$

En base al diagrama de la Figura 2.10, y al torque de control de la Ecuación 2.36, en el cual una parte se basa en el error, es decir similar a un control PD, se busca que el controlador tenga un tiempo de respuesta similar al CPC, por lo que:

$$\Lambda = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 3 \end{bmatrix}; \quad k_v = \begin{bmatrix} 12 & 0 & 0 \\ 0 & 12 & 0 \\ 0 & 0 & 12 \end{bmatrix}
 \tag{2.46}$$

Para sintonizar la matriz Γ , se han realizado simulaciones, las cuales se detallan en el Capítulo 3. Además, es importante tener en cuenta que los actuadores no pueden proporcionar un par ilimitado, puesto que la estructura mecánica como el sistema eléctrico tienen limitaciones, por lo tanto, los actuadores tienen un par máximo τ_i^{max} que pueden proporcionar, restringiéndose a:

$$|\tau_i| \leq \tau_i^{max}; \quad i = 1, 2, \dots, n \tag{2.15}$$

Siendo τ_i el par requerido por la i -ésima articulación.

Es así como se considera la saturación de manipuladores causado por las limitaciones de los actuadores, siendo este un fenómeno no lineal frecuente, que aparece cuando el algoritmo de control genera señales grandes de control conduciendo a que el actuador salga de su región de operación lineal, generando problemas como: inestabilidad, disminución del desempeño del controlador, incremento de temperatura, entre otros.

Con el fin de tomar en cuenta este efecto, se ha empleado saturadores para limitar señal de salida de cada controlador. Por lo que en este trabajo para los límites superior e inferior de estos saturadores, se ha considerado los torques máximos que de acuerdo a los datos técnicos de los servomotores que emplea la plataforma comercial, los valores máximos se muestran en la Tabla 2.1.

Tabla 2.1. Torques máximos de servomotores.

	Articulación 1 τ_1^{max}	Articulación 2 τ_2^{max}	Articulación 3 τ_3^{max}
Torque máximo (N.m)	+/-0.58	+/-2.42	+/-1.29

(Fuente: Propia)

2.5. Gestos para control del manipulador virtual

El registro de datos EMG permite visualizar la variación de la señal de cada uno de los gestos al ser realizados. Para el control y selección del objeto con el que trabajará el robot virtual, se ha considerado los siguientes gestos: wave out, wave in, close, thumb, relax e index. Cada uno de los gestos se los realizará con una ventana de 1 segundo de duración, esto debido a que después de varias pruebas realizadas se verificó, que en este lapso de tiempo la señal de cada gesto era grabada sin recortes.

Cada uno de los gestos y los registros de EMG respectivos se exponen a continuación:

2.5.1. Wave Out

El usuario gira la mano hacia afuera con los dedos juntos y alargados (Figura 2.11). El usuario puede realizar este gesto en el lapso de un segundo o más tiempo. Para este gesto se puede ver en la Figura 2.12 la señal EMG pura (con duración de un segundo a una frecuencia de muestreo de 200 Hz) y la señal EMG filtrada usando un filtro pasa bajos, a una frecuencia de 50 Hz.

La implementación del filtro es para reducir artefactos existentes a nivel de ruido no fisiológico causado por el movimiento de los sensores.



Figura 2.11. Gesto wave out, se le asocia el número "1".
(Fuente: Propia)

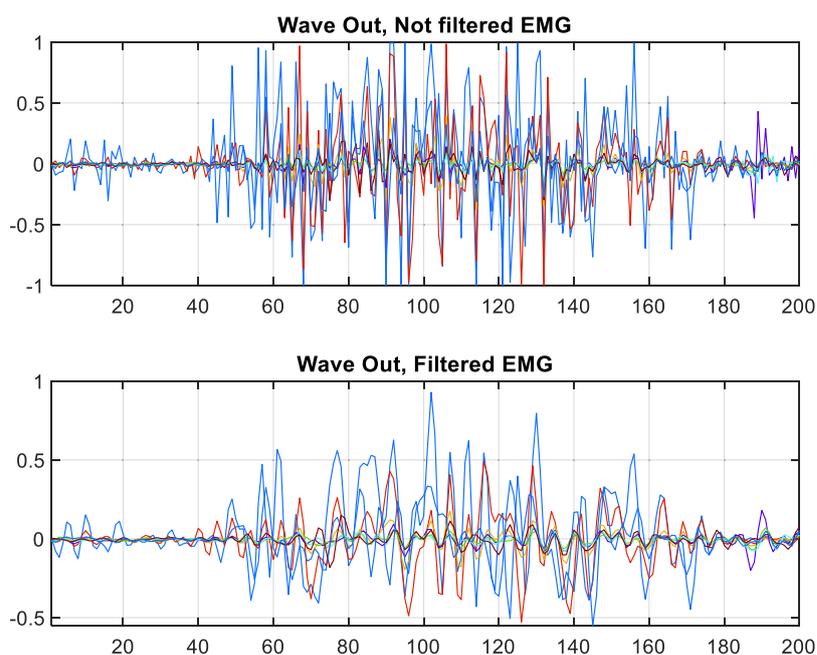


Figura 2.12. EMG del gesto wave out, señal filtrada a una frecuencia de corte de 50Hz. Se muestra 8 canales de EMG correspondientes a los 8 canales bipolares del brazalete MYO. (Fuente: Propia)

2.5.2. Wave In

El usuario gira la mano hacia adentro con los dedos juntos y alargados (Figura 2.13). El usuario puede realizar este gesto en el lapso de un segundo o más tiempo. Para este gesto se puede ver en la Figura 2.14 la señal EMG pura (con duración de un segundo a una frecuencia de muestreo de 200 Hz) y la señal EMG filtrada usando un filtro pasa bajos, a una frecuencia de 50 Hz.



Figura 2.13. Gesto wave in, se le asocia el número "2". (Fuente: Propia)

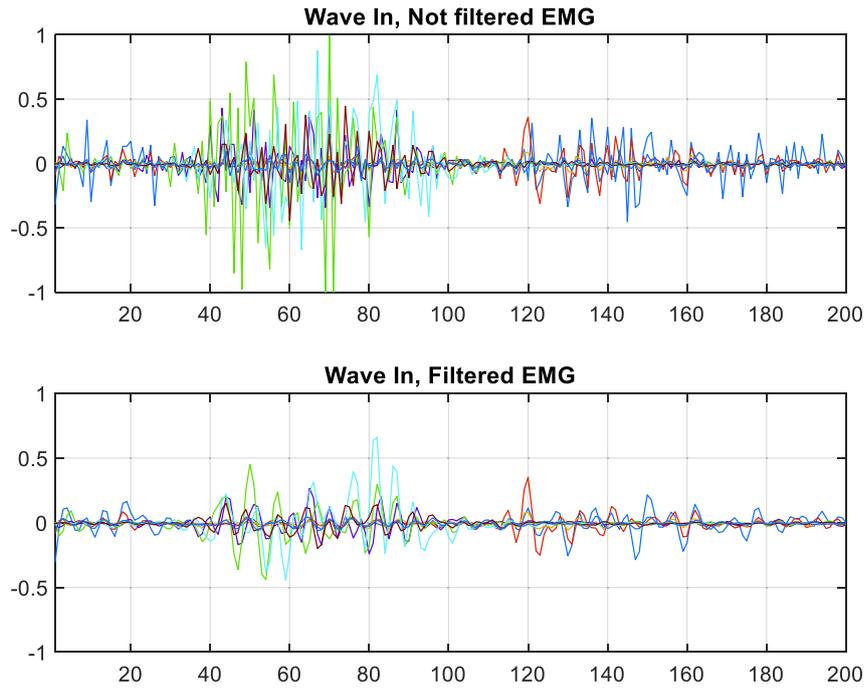


Figura 2.14. EMG del gesto wave in, señal filtrada a una frecuencia de corte de 50Hz. Se muestra 8 canales de EMG correspondientes a los 8 canales bipolares del brazalete MYO. (Fuente: Propia)

2.5.3. Close

El usuario cierra completamente su mano, realizando un puño con un poco de fuerza (Figura 2.15). El usuario puede realizar este gesto en el lapso de un segundo o más tiempo. Para este gesto se puede ver en la Figura 2.16 la señal EMG pura (con duración de un segundo a una frecuencia de muestreo de 200 Hz) y la señal EMG filtrada usando un filtro pasa bajos, a una frecuencia de 50 Hz.



Figura 2.15. Gesto close, se le asocia el número "3". (Fuente: Propia)

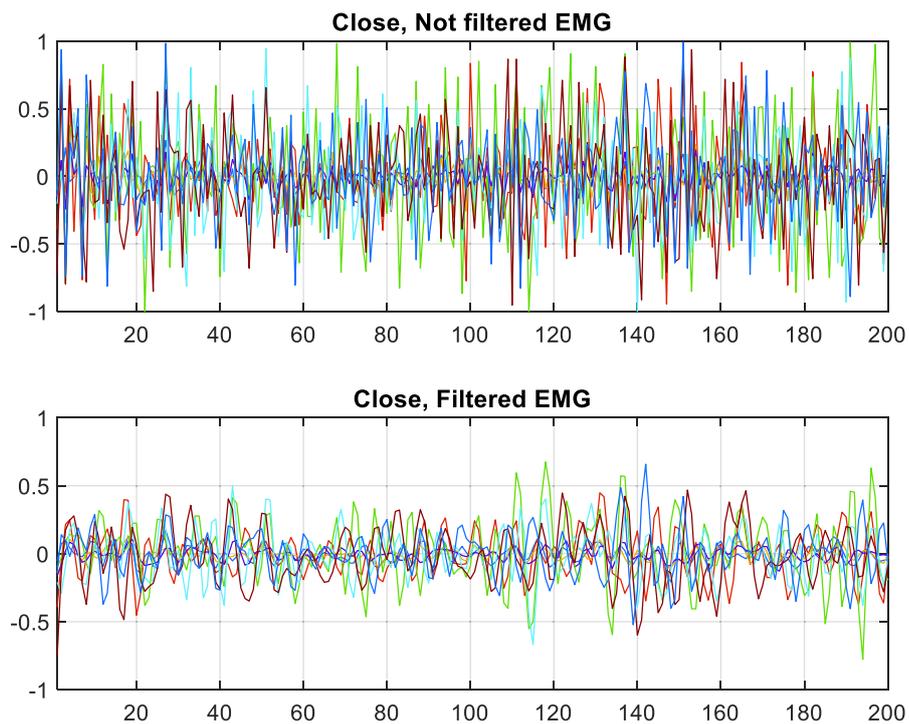


Figura 2.16. EMG del gesto close, señal filtrada a una frecuencia de corte de 50Hz. Se muestra 8 canales de EMG correspondientes a los 8 canales bipolares del brazalete MYO. (Fuente: Propia)

2.5.4. Thumb

El usuario mueve el dedo pulgar hacia arriba con el resto de los dedos en forma cerrada (Figura 2.17). El usuario puede realizar este gesto en el lapso de un segundo o más tiempo. Para este gesto se puede ver en la Figura 2.18 la señal EMG pura (con duración de un segundo a una frecuencia de muestreo de 200 Hz) y la señal EMG filtrada usando un filtro pasa bajos, a una frecuencia de 50 Hz.



Figura 2.17. Gesto thumb, se le asocia el número "5". (Fuente: Propia)

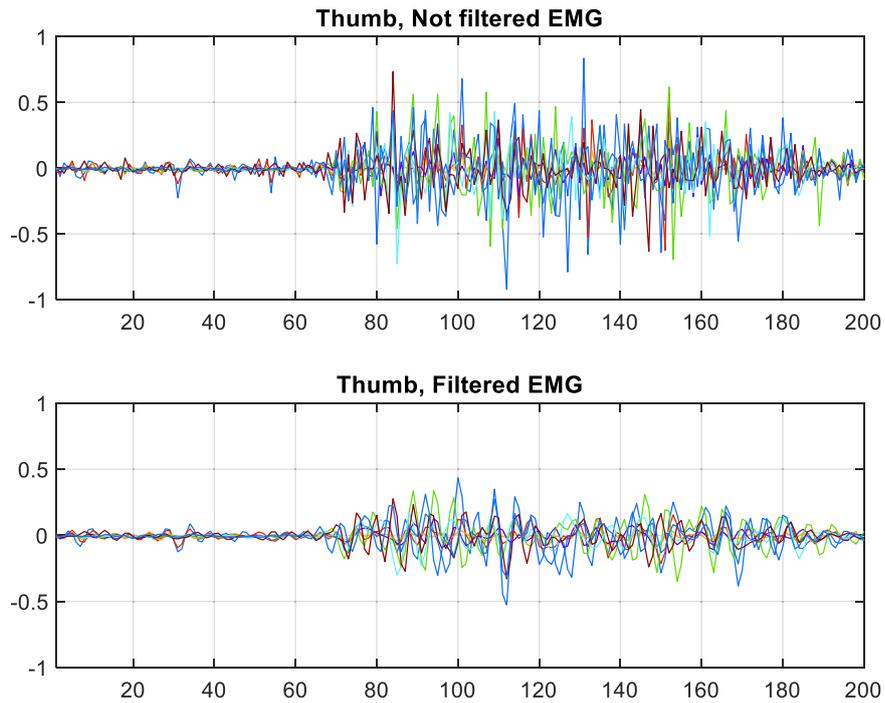


Figura 2.18. EMG del gesto thumb, señal filtrada a una frecuencia de corte de 50Hz. Se muestra 8 canales de EMG correspondientes a los 8 canales bipolares del brazalete MYO. (Fuente: Propia)

2.5.5. Relax

El usuario no realiza ningún gesto con su mano o el movimiento es leve (Figura 2.19). El usuario puede realizar este gesto en el lapso de un segundo o más tiempo. Para este gesto se puede ver en la Figura 2.20 la señal EMG pura (con duración de un segundo a una frecuencia de muestreo de 200 Hz) y la señal EMG filtrada usando un filtro pasa bajos, a una frecuencia de 50 Hz.



Figura 2.19. Gesto relax, se le asocia el número "6". (Fuente: Propia)

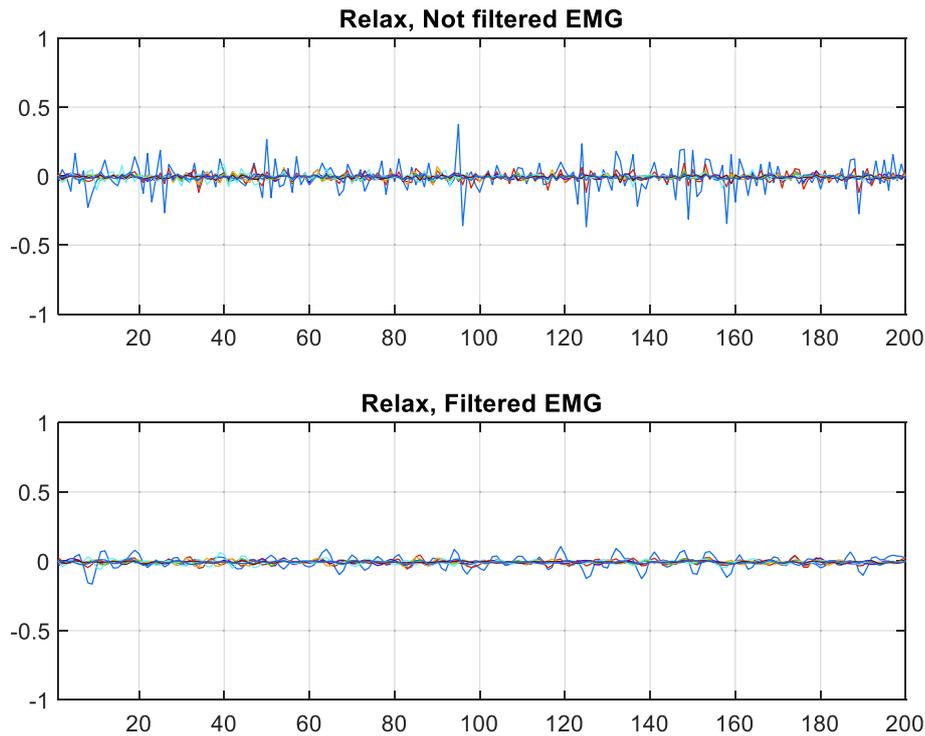


Figura 2.20. EMG del gesto relax, señal filtrada a una frecuencia de corte de 50Hz. Se muestra 8 canales de EMG correspondientes a los 8 canales bipolares del brazalete MYO. (Fuente: Propia)

2.5.6. Index

El usuario mueve el dedo índice hacia afuera con el resto de los dedos en forma cerrada (Figura 2.21). El usuario puede realizar este gesto en el lapso de un segundo o más tiempo. Para este gesto se puede ver en la Figura 2.22 la señal EMG pura (con duración de un segundo a una frecuencia de muestreo de 200 Hz) y la señal EMG filtrada usando un filtro pasa bajos, a una frecuencia de 50 Hz.



Figura 2.21. Gesto index, se le asocia el número "7". (Fuente: Propia)

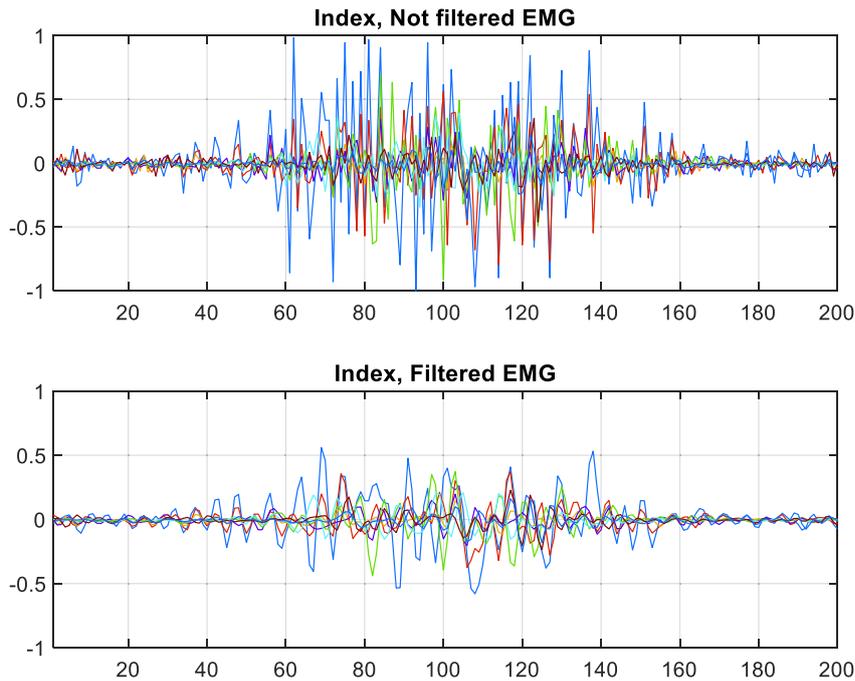


Figura 2.22. EMG del gesto index, señal filtrada a una frecuencia de corte de 50Hz. Se muestra 8 canales de EMG correspondientes a los 8 canales bipolares del brazalete MYO. (Fuente: Propia)

2.6. Medidas estadísticas y extracción de características

Este módulo es el responsable de extraer características de cada gesto, que sean no redundantes y que sean lo suficientemente discriminativos entre cada gesto para usarlos en el módulo de clasificación. Las medidas estadísticas empleadas sobre las señales EMG se utilizan para extraer características. Las características que se usa son el valor medio, desviación estándar, la envolvente de la curva EMG, ventaneo, energía, valor medio absoluto. Estas características son aplicadas a cada canal del brazalete por lo que por cada característica tendremos un vector de 8 columnas por 1 fila.

2.6.1. Valor medio

Este es un estimador del valor medio de la señal de entrada filtrada que tiene una longitud de N muestras. Este puede ser expresado de la siguiente forma por cada canal:

$$\mu = \frac{1}{N} \sum_{i=1}^N V_i \quad (2.47)$$

$$V_i \in [-1, 1]$$

Donde:

- ✓ V_i , valores de la señal EMG en el instante i .
- ✓ μ , valor medio.
- ✓ N , número de datos, $N=200$.

2.6.2. Desviación estándar

La desviación estándar mide la dispersión de un conjunto de datos (o población). La desviación estándar es la medida de dispersión más común, que indica qué tan dispersos están los datos con respecto a la media. Mientras mayor sea la desviación estándar, mayor será la dispersión de los datos. La expresión siguiente muestra la desviación estándar por cada canal.

$$S = \sqrt{\frac{1}{N-1} \sum_{i=1}^N |X_i - \mu|^2} \quad (2.48)$$

$X_i \in [-1, 1]$

Donde:

- ✓ X_i , valores de la señal EMG en el instante i .
- ✓ μ , valor medio.
- ✓ S , desviación estándar.
- ✓ N , número de datos, $N=200$.

2.6.3. Envoltente de la curva EMG

La transformada de Hilbert es una herramienta matemática útil para describir la envoltente compleja de una señal modulada con una portadora real. Su definición es:

$$H\{f(t)\} = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{f(u)}{u-t} du \quad (2.49)$$

La transformada de Hilbert facilita la formación de la señal analítica. La envoltente de una función es definida como el módulo de su función analítica (Figura 2.23).

$$E(t) = |g(t)| = \sqrt{f(t)^2 + (H\{f(t)\})^2} \quad (2.50)$$

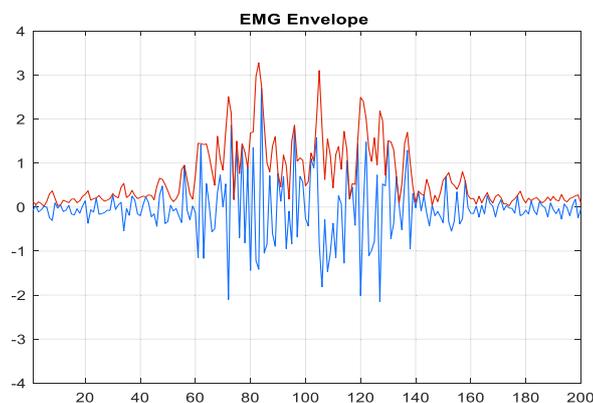


Figura 2.23. Envoltente de la curva EMG (color naranja).
(Fuente: Propia)

2.6.4. Cruces por cero

Los Cruces por cero indican el número de veces que una señal continua toma el valor de cero. Para las señales discretas, un cruce por cero ocurre cuando dos muestras consecutivas difieren de signo, o bien una muestra toma el valor de cero. Consecuentemente, las señales con mayor frecuencia presentan un mayor valor de esta característica, el ruido también genera un gran número de cruces por cero [47].

$$z = \sum_{i=0}^{N-1} | \text{sign}[x(i)] - \text{sign}[x(i-1)] | \quad (2.51)$$

2.6.5. Energía

Básicamente este algoritmo es similar al de los cruces por cero, pero a las funciones signo, se les ha multiplicado por la energía de la muestra analizada, es así que la densidad acumulada no sólo dependerá del cambio de signo de las muestras sino también de su amplitud, por esto, el ruido no logrará una gran acumulación de densidad, debido a que posee una pequeña amplitud comparada con el gesto realizado [47]. Su formulación matemática se muestra a continuación:

$$E = \sum_{i=0}^N | y[i] * y[i] * \text{sign}[x(i)] - y[i-1] * y[i-1] * \text{sign}[x(i-1)] | \quad (2.52)$$

2.6.6. Densidad espectral de potencia

Esta función toma un vector de cada canal EMG, y lo divide en segmentos más pequeños para su análisis. A medida que aumenta el número de segmentos disponibles, la función producirá un espectro de potencia más suave con menor variación y con valores de potencia más cercanos a los valores esperados. Para el cálculo de la densidad espectral de potencia se usa una ventana BlackmanHarris de 200 puntos.

2.6.7. Valor medio absoluto

Este parámetro permite obtener un valor positivo promedio por cada canal analizado. Dependiendo del gesto, el valor promedio absoluto en cada canal es diferente y es similar siempre que se realice un movimiento similar.

$$\mu = \frac{1}{N} \sum_{i=1}^N |V_i| \quad (2.53)$$

$V_i \in [-1, 1]$

2.7. Algoritmos de clasificación

Previo al uso de los algoritmos de clasificación tenemos que organizar los datos de cada canal para poder tener un mejor resultado en el entrenamiento de los algoritmos de clasificación. La Figura 2.24 muestra una gráfica de los datos organizados y pre procesado para cada uno de los gestos realizados en este proyecto.

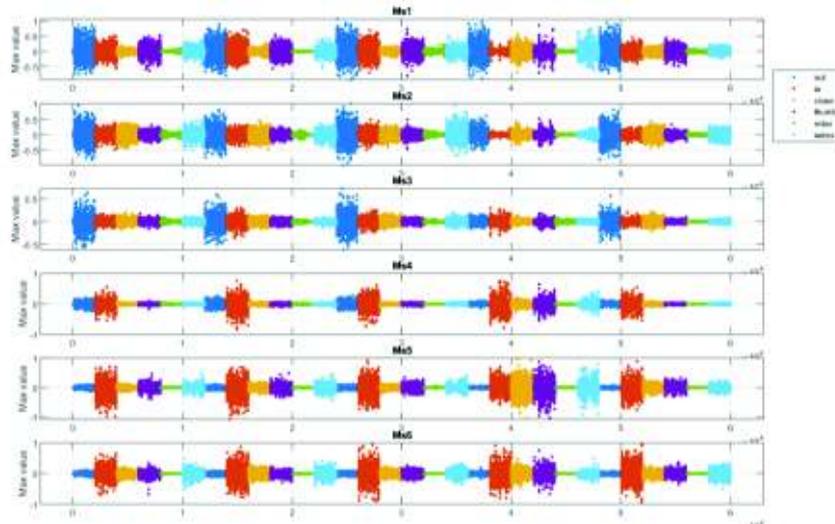


Figura 2.24. EMG organizadas por sensor.
(Fuente: Propia)

Adicional, para organizar los datos se ha tomado como referencia las recomendaciones del fabricante de tratar de colocar el brazalete en la misma posición, sin embargo, esto no es posible siempre. Para cada grabación de los gestos se ha registrado el sensor ubicando en una posición que se definirá más adelante como posición de referencia al momento de grabar las señales.

Para el proyecto se analizó los resultados obtenidos para los clasificadores mostrados a continuación y se seleccionó como idóneo el que tiene mayor porcentaje sin tomar mucho en consideración el tiempo de respuesta ya que todos los clasificadores analizados tienen una respuesta menos a los 300ms.

2.7.1. Máquina de vectores de soporte (SVM)

Las SVM implementan algoritmos de clasificación con aprendizaje supervisado y se asume que se parte con n datos de entrenamiento N dimensionales x_i , siendo x el vector de características que representa al objeto i con su respectiva etiqueta y_i .

En general el planteamiento original de las SVM se centró en resolver problemas de clasificación binaria, donde el objetivo es generar un modelo capaz de separar los objetos de dos clases distintas por medio de un hiperplano óptimo, y en donde la solución puede

ser escrita por la combinación de unos pocos puntos de entrada que son los llamados vectores de soporte. Para los casos en que las clases son linealmente separables el hiperplano se calcula en el espacio original. Sin embargo, para el caso de discriminar gestos de la mano esta linealidad no existe; en tales casos, las SVM aplican una transformación del espacio de entrada a un espacio multidimensional en el cual las clases (etiquetas) si son linealmente separables [40].

Esta transformación no lineal se realiza de distintas maneras, empleando mapeos, también llamados Kernels, tales como los Polinomiales y RBF. En la aplicación desarrollada para el reconocimiento de los 6 gestos se usa un kernel polinomial. El kernel polinomial no solo observa las características dadas de las muestras de entrada para determinar su similitud, sino también las combinaciones de éstas.

2.7.2. Vecinos más próximos (KNN)

El método de los k vecinos más cercanos (en inglés, k-nearest neighbors) es un método de clasificación supervisada (Aprendizaje, estimación basada en un conjunto de entrenamiento y prototipos) que sirve para estimar la función de densidad de las predictoras por cada clase [27].

2.7.3. Árbol de decisiones

Un árbol de decisión es una representación simple para clasificar ejemplos. El aprendizaje basado en árboles de decisión es una de las técnicas más eficaces para la clasificación supervisada porque la predicción es rápida, es fácil de interpretar, analiza el problema para que todas las opciones sean consideradas. Para esto se supone que todas las funciones tienen dominios discretos finitos, y existe una sola característica de destino llamado la clasificación. Cada elemento del dominio de la clasificación se llama clase. Un árbol de decisión o un árbol de clasificación es un árbol en el que cada nodo interno está etiquetado con una función de entrada. Los arcos procedentes de un nodo etiquetado con una característica están etiquetados con cada uno de los posibles valores de la característica. Cada hoja del árbol se marca con una clase o una distribución de probabilidad sobre las clases [41].

2.8. Matrices de datos de entrenamiento y testeo

Los datos de entrenamiento como de testeo son los que se utilizan para verificar el funcionamiento y entrenamiento del clasificador. Cada usuario realiza 6 gestos y por cada gesto se realizan 10 repeticiones, dando un total de 60 gestos por cada usuario. Se toma datos de 6 usuarios para la base de entrenamiento y 6 usuarios para la base de datos de

testeo. Cada gesto se lo realiza en un lapso de 1 segundo de duración, y cada gesto tiene una matriz de 8 columnas y 200 filas.

$$\begin{aligned}
 & \text{User 1:} \\
 & \text{Data} = [Ms1, Ms2, \dots, Msn] \\
 & Msn = Matrix_{10 \times 200} / \text{Category} \\
 & \quad n \in [1, 8] \\
 & \text{Category} = [\text{out}, \text{in}, \text{close}, \text{thumb}, \text{relax}, \text{index}] \\
 & \text{Dtr}_1 = [Ms1_{60 \times 200}, Ms2_{60 \times 200}, \\
 & \quad Ms3_{60 \times 200}, Ms4_{60 \times 200}, \\
 & \quad Ms5_{60 \times 200}, Ms6_{60 \times 200}, \\
 & \quad Ms7_{60 \times 200}, Ms8_{60 \times 200}]
 \end{aligned} \tag{2.54}$$

Data almacena los registros EMG por cada canal y por cada gesto. **Ms1** corresponde a la matriz del sensor 1.

Dtr1 es la matriz que contiene todos los datos de entrenamiento por cada canal y por cada repetición.

El brazalete MYO tiene 8 sensores *s1* hasta *s8*, se extrae cada columna de cada matriz y de cada gesto. Se obtiene una matriz de entrenamiento por persona *Dtr*. A cada tabla organizada de la misma manera se le aplica los análisis estadísticos mencionados. Una tabla general y completa se obtiene de los 6 usuarios. Para el usuario 1 los datos de entrenamiento se grabaron usando como referencia el sensor 1 del brazalete, lo que quiere decir que al momento de grabar las señales EMG el sensor 1 (*s1*) coincidía con la posición sugerida por el fabricante (*s4*). Esto con el fin de que al momento de ingresar los datos para el entrenamiento se debe organizarlos con el mismo punto de referencia:

$$\begin{aligned}
 & M_{usr1} = Matrix_{60 \times 200} \\
 & D_{usr1} = [Ms1_{60 \times 200}, Ms2_{60 \times 200}, \\
 & \quad Ms3_{60 \times 200}, Ms4_{60 \times 200}, \\
 & \quad Ms5_{60 \times 200}, Ms6_{60 \times 200}, \\
 & \quad Ms7_{60 \times 200}, Ms8_{60 \times 200}]
 \end{aligned} \tag{2.55}$$

Musr1 matriz completa del usuario 1, comprende las 10 repeticiones por cada gesto (6 gestos en total), con una duración de un segundo (200 puntos).

Para el usuario 2 se usó el sensor 5 (*s5*) como sensor de referencia cuando se realizó la grabación de los gestos (Figura 2.25).

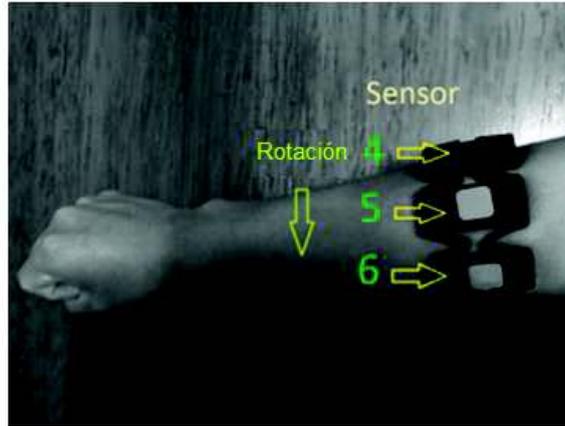


Figura 2.25. Sensor de referencia para usuario 2.
(Fuente: Propia)

Matriz de datos de entrenamiento para el usuario 2:

$$\begin{aligned}
 M_{usr2} &= Matrix_{60 \times 200} \\
 D_{usr2} &= [Ms1_{60 \times 200}, Ms2_{60 \times 200}, \\
 & Ms3_{60 \times 200}, Ms4_{60 \times 200}, \\
 & \boxed{Ms5}_{60 \times 200}, Ms6_{60 \times 200}, \\
 & Ms7_{60 \times 200}, Ms8_{60 \times 200}]
 \end{aligned} \tag{2.56}$$

Para el usuario 3, el brazalete se rotó nuevamente tomando como referencia el sensor 3 (s3).

$$\begin{aligned}
 M_{usr3} &= Matrix_{60 \times 200} \\
 D_{usr3} &= [Ms1_{60 \times 200}, Ms2_{60 \times 200}, \\
 & \boxed{Ms3}_{60 \times 200}, Ms4_{60 \times 200}, \\
 & Ms5_{60 \times 200}, Ms6_{60 \times 200}, \\
 & Ms7_{60 \times 200}, Ms8_{60 \times 200}]
 \end{aligned} \tag{2.57}$$

Para los usuarios 4, 5, 6 los sensores usados como referencia son el s3, s8, s6 respectivamente. La matriz de entrenamiento de todos los usuarios se obtiene de:

$$\begin{aligned}
 Data &= [Ms1, Ms2, \dots, Msn] \\
 Msn &= Matrix_{60 \times 200} \\
 & n \in [1, 8] \\
 Category &= \\
 & [out, in, close, thumb, relax, index] \\
 Dtr_m &= [Ms1_{360 \times 200}, Ms2_{360 \times 200}, \\
 & Ms3_{360 \times 200}, Ms4_{360 \times 200}, \\
 & Ms5_{360 \times 200}, Ms6_{360 \times 200}, \\
 & Ms7_{360 \times 200}, Ms8_{360 \times 200}]
 \end{aligned} \tag{2.58}$$

$$\begin{aligned}
 DefaultMatrix &= [s1, s2, s3, s4, s5, s6, s7, s8] \\
 s1 &= sensor1
 \end{aligned}$$

Para la matriz de testeo se tomó como referencia el *sensor4* para todos los usuarios. La matriz de cada usuario es similar a la matriz de entrenamiento con la diferencia que para los datos de testeo se tomó como referencia únicamente el sensor 4 para todos los usuarios.

Matriz por usuario

$$Dts = [(W1, E1), \dots, (Wn, En)]$$

$$Dts = Matrix_{200 \times 8}, Wi \in [-1, 1]_{200 \times 1}, Ei \in [1, 8] \quad (2.59)$$

Matriz total de testeo

$$M_{6users} = Matrix_{360 \times 200}$$

$$D_{6users} = [Ms1_{360 \times 200}, Ms2_{360 \times 200},$$

$$Ms3_{360 \times 200}, Ms4_{360 \times 200},$$

$$Ms5_{360 \times 200}, Ms6_{360 \times 200},$$

$$Ms7_{360 \times 200}, Ms8_{360 \times 200}] \quad (2.60)$$

Cada una de las matrices entrenamiento, testeo pasan por cada función extractora de características en forma de tablas para agilizar el procesamiento global.

2.9. Matlab Classification Learner

Esta aplicación permite ingresar la matriz de datos con cada etiqueta correspondiente y organizada por cada canal. Para el entrenamiento se separa un porcentaje de los datos de entrenamiento para testear el sistema después de entrenarlo y validar cada uno de los clasificadores mencionados anteriormente.

En la aplicación Classification Learner de Matlab la opción Holdout Validation (Figura 2.26) permite usar un porcentaje de los datos de la matriz ingresada para realizar las pruebas al algoritmo de reconocimiento durante el entrenamiento.

Classification Learner es una aplicación de Matlab que permite entrenar algoritmos de clasificación de forma rápida y presenta diferentes alternativas de clasificadores. Se ingresan los datos a la aplicación en forma de tabla asignándole su respectiva etiqueta.

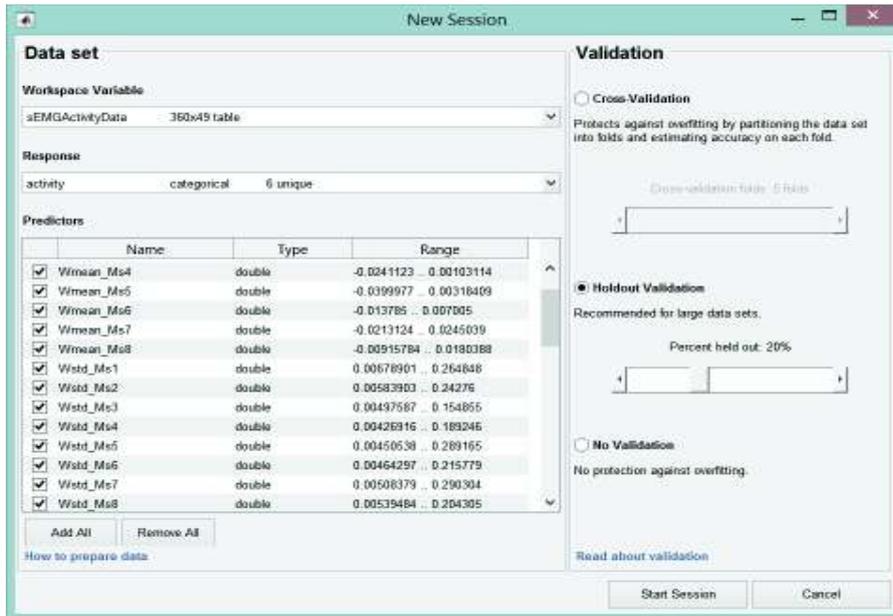


Figura 2.26. Opción para pruebas, Holdout Validation.
(Fuente: Propia)

En la Figura 2.27 se muestra un breve resumen de las características para 7 repeticiones del gesto Wave Out correspondientes al canal 1 y al canal 8.

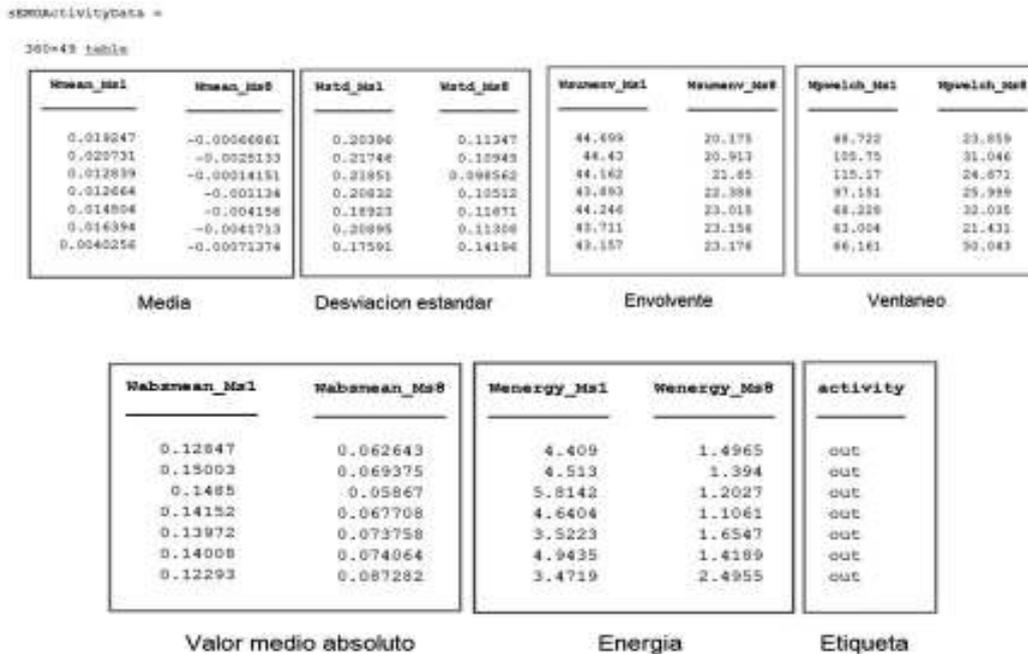


Figura 2.27. Características ingresadas para la clasificación de gesto.
(Fuente: Propia)

Usando la aplicación de Matlab se procede a ingresar el vector de las características de cada gesto al sistema y entrenamos diferentes clasificadores, SVM, KNN, Árbol de decisiones y se verifica la exactitud del clasificador, así como los tiempos de respuesta de éstos clasificadores.

2.9.1. Entrenamiento

2.9.1.1. Clasificador SVM

La Figura 2.28 muestra la matriz de confusión usando el clasificador SVM para el conjunto de datos de entrenamiento con un Holdout del 20%.

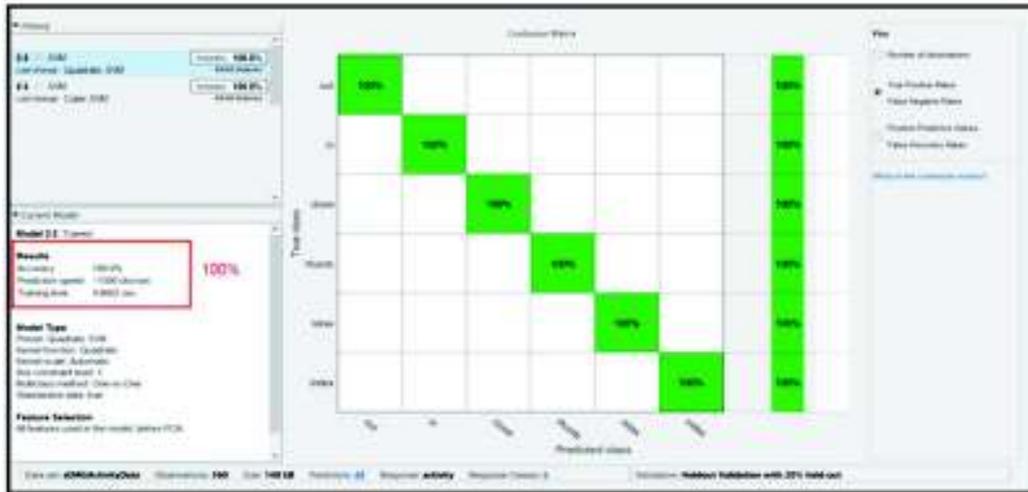


Figura 2.28. Matriz de confusión con una exactitud del 100%, clasificador SVM usando la base de datos de entrenamiento.
(Fuente: Propia)

Tiempo promedio de respuesta del clasificador SVM para las señales de entrenamiento 18.54 ms. El tiempo de respuesta se mide desde que se ingresa los datos pre procesados hasta cuando el clasificador da una respuesta.

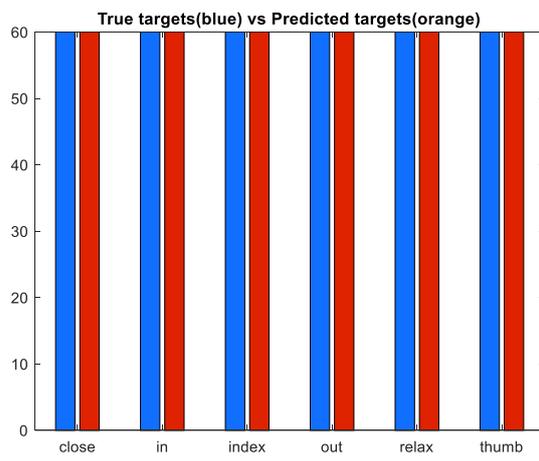


Figura 2.29. Respuesta obtenida vs cantidad de repeticiones de gestos, 6 usuarios, SVM clasificador.
(Fuente: Propia)

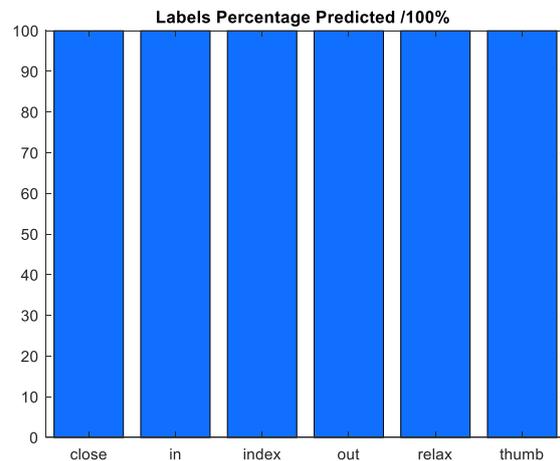


Figura 2.30. Porcentaje según el gesto usando SVM para los 6 usuarios.
(Fuente: Propia)

2.9.1.2. Clasificador KNN

La Figura 2.31 muestra la matriz de confusión usando el clasificador KNN para el conjunto de datos de entrenamiento con un Holdout del 20%.

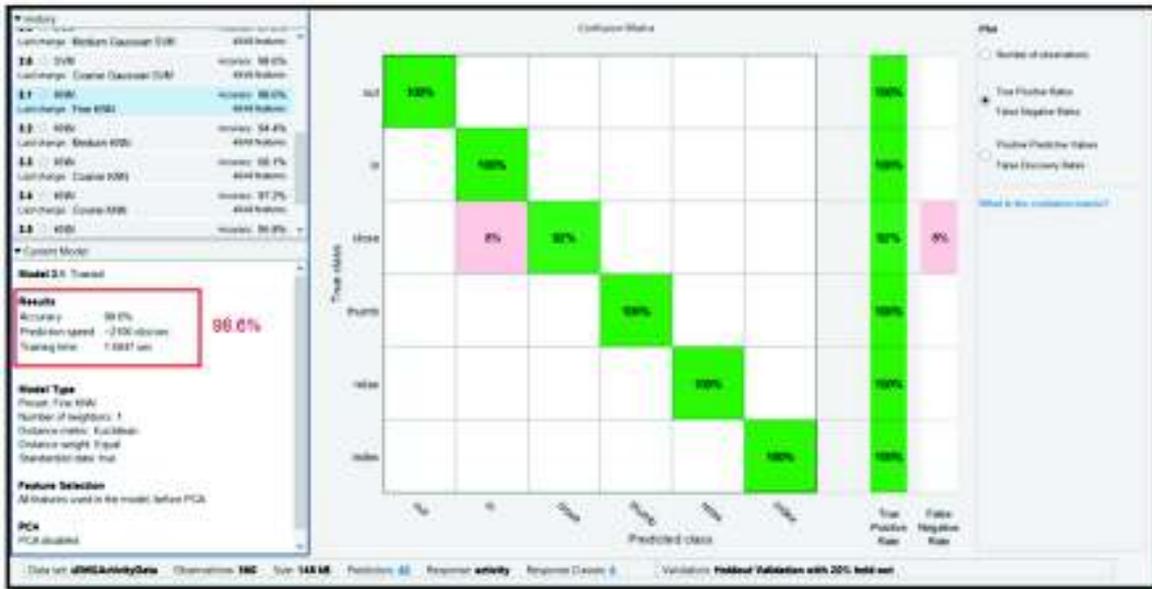


Figura 2.31. Matriz de confusión con una exactitud del 98.6%, clasificador KNN usando la base de datos de entrenamiento.
(Fuente: Propia)

Tiempo promedio de respuesta del clasificador KNN para las señales de entrenamiento: 11.67 ms. El tiempo se mide desde que se ingresa los datos pre procesados hasta cuando el clasificador da una respuesta.

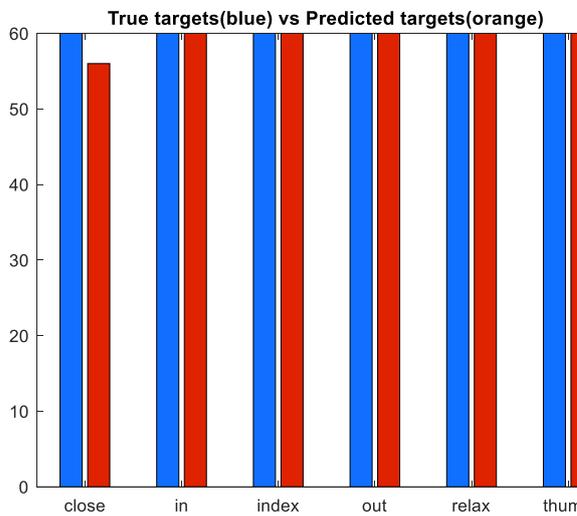


Figura 2.32. Respuesta obtenida vs cantidad de repeticiones de gestos, 6 usuarios, KNN clasificador.
(Fuente: Propia)

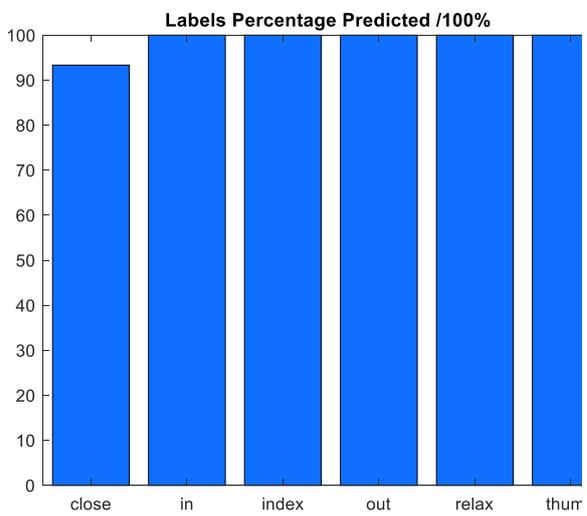


Figura 2.33. Porcentaje según el gesto usando KNN para los 6 usuarios.
(Fuente: Propia)

2.9.1.3. Clasificador Árbol de decisiones

La Figura 2.34 muestra la matriz de confusión usando el clasificador Árbol de decisiones para el conjunto de datos de entrenamiento con un Holdout del 20%.

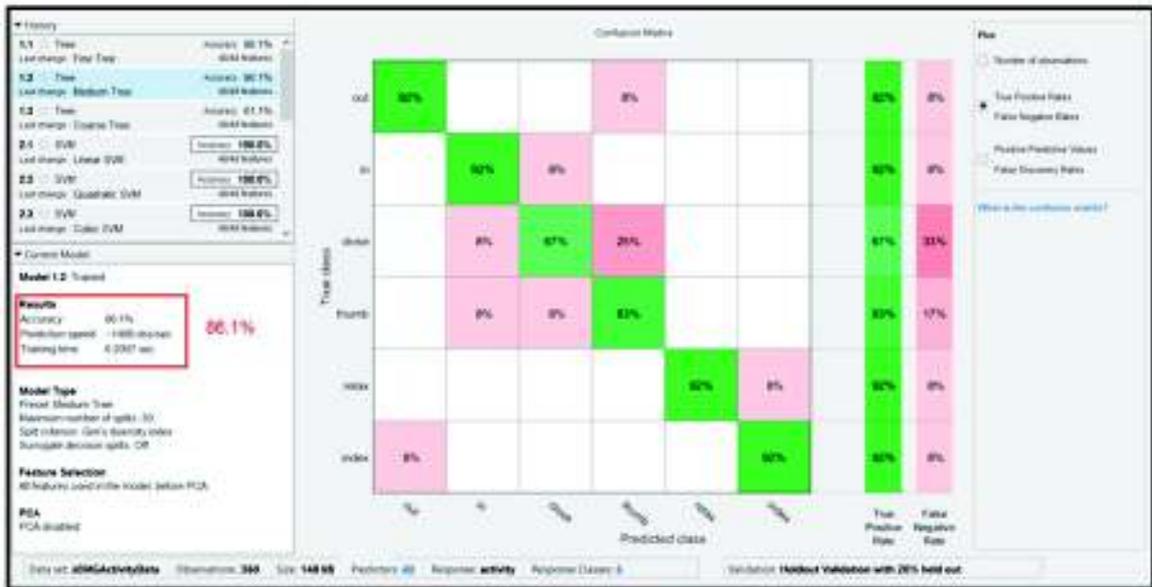


Figura 2.34. Matriz de confusión con una exactitud del 86.1%, clasificador árbol de decisiones usando la base de datos de entrenamiento. (Fuente: Propia)

Tiempo promedio de respuesta del clasificador Árbol de decisiones para las señales de entrenamiento: 10.07 ms. El tiempo se mide desde que se ingresa los datos pre procesados hasta cuando el clasificador da una respuesta.

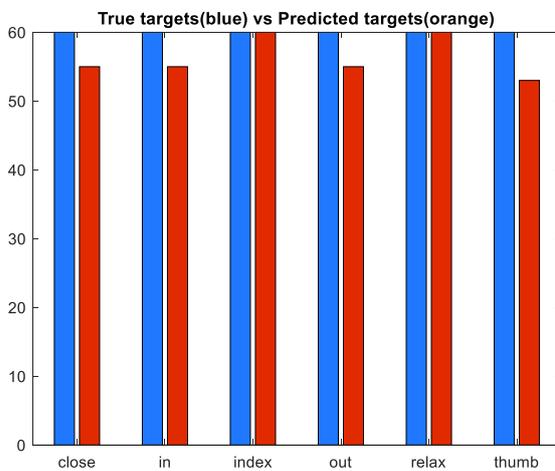


Figura 2.35. Respuesta obtenida vs cantidad de repeticiones de gestos, 6 usuarios, clasificador árbol de decisión. (Fuente: Propia)

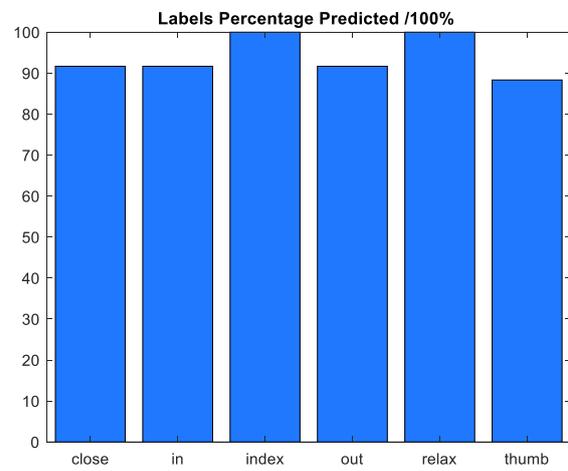


Figura 2.36. Porcentaje según el gesto usando árbol de decisión para los 6 usuarios. (Fuente: Propia)

Se puede ver que para cada clasificador ingresando los mismos datos se obtienen diferentes porcentajes de acierto, es así que el clasificador árbol de decisiones a pesar de ser muy rápido tiene bajo porcentaje de acierto comparado con los otros dos.

De los clasificadores probados con los datos de entrenamiento se ha seleccionado el clasificador SVM ya que este presenta un mayor porcentaje de acierto con un tiempo de respuesta bajo los 300ms para que se pueda considerar como tiempo real [48].

2.10. HMI de control del brazo robótico

2.10.1. Interfaz principal (MYO 1)

La interfaz desarrollada para el control del brazo robótico presenta varias pestañas que se describen a continuación.

2.10.1.1. Ventana de calibración

Se diseñó una ventana de calibración (Figura 2.37), la cual sirve para setear el sistema y poder manejar ordenadamente los datos de la matriz que se obtiene al realizar cada uno de los 6 gestos, *wave out*, *wave in*, *close*, *thumb*, *relax*, *index*.

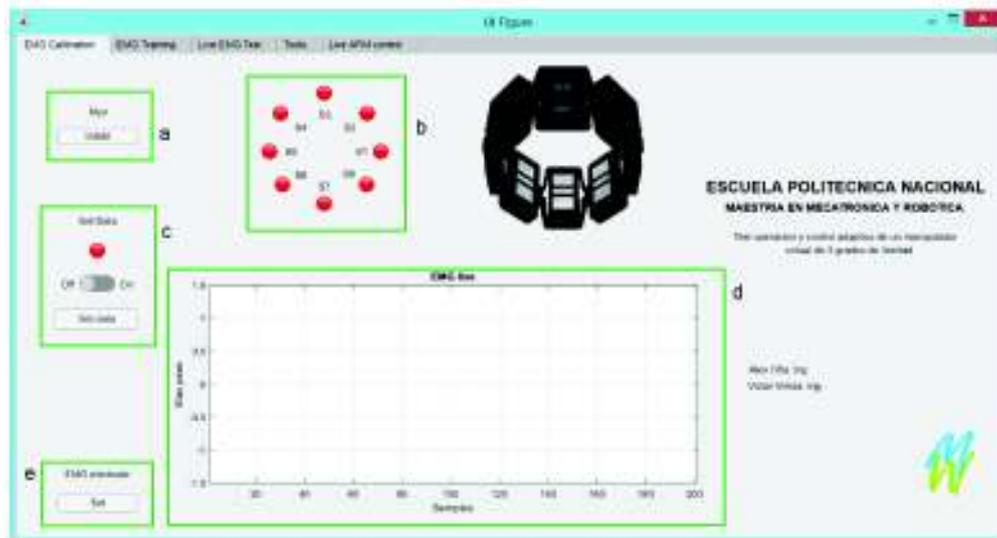


Figura 2.37. Interfaz MYO 1, pestaña de calibración. a) Instalar Myo, b) Indicador de sensor de referencia, c) Inicio de adquisición, d) Graficación EMG, e) Seteo sensor de referencia. (Fuente: Propia)

Para esta interfaz se identifica a cada uno de los sensores del MYO (Figura 2.38).



Figura 2.38. Distribución de los sensores en el brazalete MYO. (Fuente: Propia)

2.10.1.2. Ventana de entrenamiento

En la Figura 2.39 se muestra la ventana para el entrenamiento del sistema. En donde se puede ver el tiempo (4 segundos) que sirve para poder realizar cada uno de los gestos después de presionar en el botón REC, este tiempo es para alertar con anticipación al usuario para que realice el gesto. Cada vez que se realiza una grabación se presiona la tecla SAVE, en el caso que la grabación sea correcta y dentro de la ventada de grabación de 1 segundo. En el caso que la grabación no esté dentro de la ventana se puede eliminar la grabación presionando la tecla DELETE.

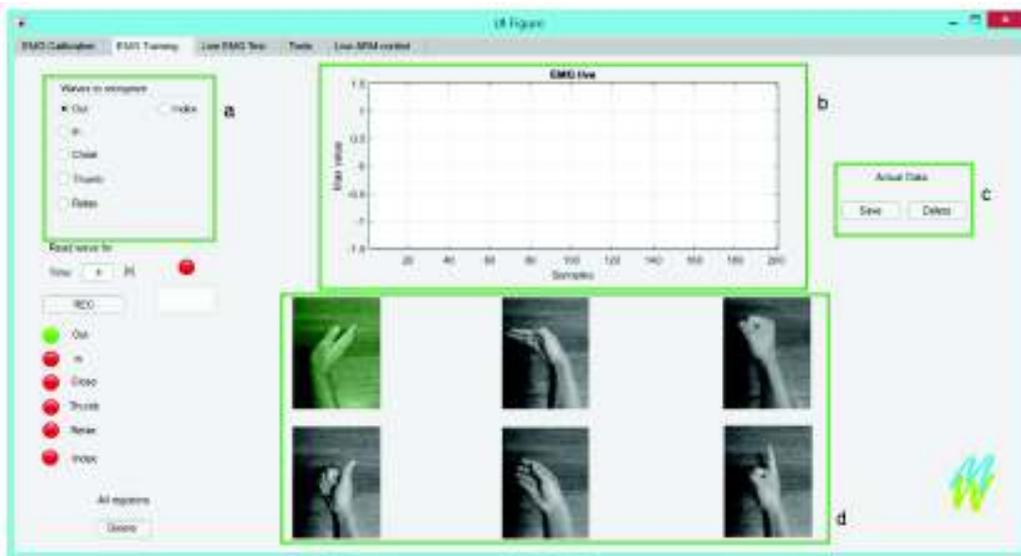


Figura 2.39. Interfaz MYO 1, pestaña de entrenamiento. a) Selección del gesto a entrenar, b) Graficación EMG, c) Guardar datos adquiridos, d) Gesto actual de entrenamiento. (Fuente: Propia)

2.10.1.3. Ventana de test

Esta ventana (Figura 2.40) se diseñó para realizar un test previo al control del sistema de reconocimiento de gestos. En tiempo real se realiza distintos movimientos con la mano, cada gesto activa una luz que cambia de color rojo a verde cuando un gesto es reconocido. Por default la luz que permanece encendida es la de relajación, cuando no se realiza ningún gesto.

En esta ventana se grafican las señales EMG mientras se realiza un gesto y se reconoce el mismo; esto con el propósito de estar seguros de que el gesto realizado está en la ventana de trabajo de 1 segundo. El sistema de reconocimiento también responde correctamente si el gesto se mantiene en el transcurso del tiempo (gesto mapeado constantemente).

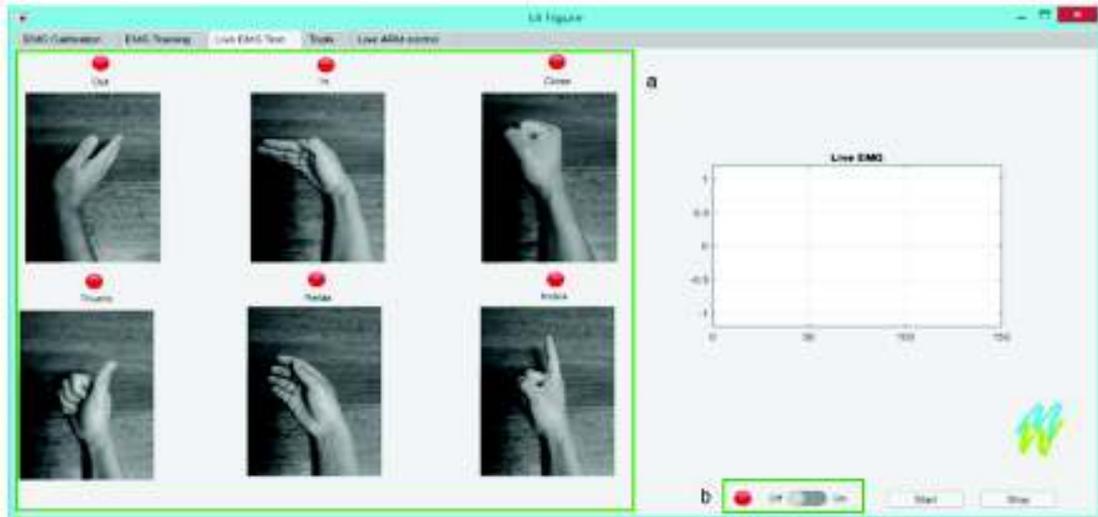


Figura 2.40. Interfaz MYO 1, pestaña de testeo. a) Gesto detectado, b) Habilitar el reconocimiento de gestos.
(Fuente: Propia)

2.10.1.4. Ventana selección de objeto y color

En esta ventana (Figura 2.41) se puede seleccionar la masa del objeto que el brazo virtual va a mover, así como también el color de éste. La selección de la masa o el color del objeto a mover se lo realiza con el reconocimiento de gestos.



Figura 2.41. Interfaz MYO 1, pestaña de selección de herramientas y colores. a) Indicador de la característica actual, b) Muestra el peso y color seleccionado.
(Fuente: Propia)

Con el gesto *wave out* se mueve automáticamente el indicador luminoso hacia la derecha, con el gesto *wave in* se puede mover el indicador hacia la izquierda. Los indicadores luminosos se activan una vez el programa este corriendo y cambian de color de rojo a verde.

Por default se ha ubicado el indicador en la primera posición superior para moverlo de acuerdo al gesto realizado. Si se sostiene en el tiempo los gestos *wave out* o *wave in*, automáticamente el indicador luminoso se mueve de forma continua hacia la derecha o hacia la izquierda respectivamente. Para seleccionar el objeto o color se usa el gesto *thumb*.

2.10.1.5. Conexión remota y control del brazo robótico

Datos de color y peso de la herramienta de trabajo junto con datos de las 2 IMU's son enviados a un computador 2 para el control del manipulador. El total de variables que viajan remotamente por la red inalámbrica consta de los siguientes datos y que se actualizan constantemente para poder controlar el brazo en tiempo real.

- Señal gesto
- Peso objeto
- Color objeto
- Pitch_my01
- Yaw_my01
- Roll_my02
- Pitch_my02
- Yaw_my02

Para el proyecto se usan 2 computadores, el computador principal donde están conectados los brazaletes MYO 1 y MYO 2. Desde este computador principal se envía los datos procesados tanto de las IMU así como de los datos EMG hacia el computador 2 donde corre la simulación del manipulador.

Usando el comando *tcpip* de Matlab se establece la conexión remota entre los computadores que se encuentran en una misma red de área local WLAN.

En la interfaz principal el funcionamiento es como cliente y como servidor pues recibe datos del brazaletes MYO 2 y también envía el paquete completo remotamente para el control del brazo robótico (Figura 2.42).

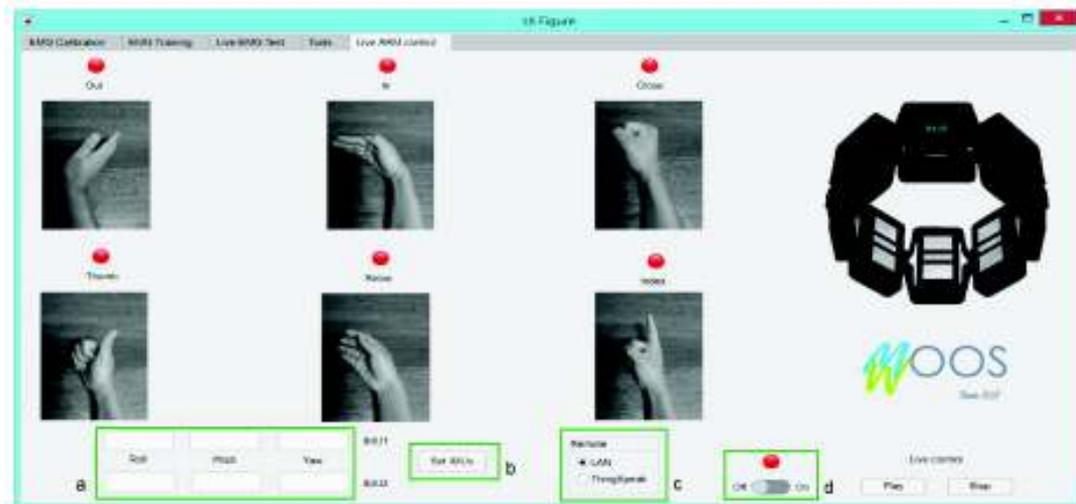


Figura 2.42. Interfaz MYO 1, pestaña de conexión remota y control del brazo robótico. a) Posición actual del brazo, b) Seteo del punto de referencia de los brazaletes, c) Tipo de conexión con el brazo virtual, d) Adquisición de datos IMU del brazaletes 2.
(Fuente: Propia)

2.10.1.6. Procesamiento de datos del sensor IMU

Para el procesamiento se actualiza la lectura de la matriz de cuaterniones cada 200ms. Este tiempo se estableció mediante pruebas propias para permitir que el computador 2 pueda recibir los datos correctamente. Los datos del sensor IMU se los almacena en un vector y se lee valor por valor para diferenciar cada uno de los datos del cuaternión.

Para tomar una posición como referencia se usa la multiplicación de cuaterniones, se toma como referencia un punto del espacio (persona sentada, con el brazo pegado al cuerpo, formando un ángulo de 90 grados) y se guarda como referencia. Para obtener la posición relativa a ese punto se multiplica el cuaternión actual con el inverso del cuaternión tomado como referencia.

$$q = q_0 + \mathbf{q} = q_0 + q_1\hat{i} + q_2\hat{j} + q_3\hat{k}. \quad (2.61)$$

$$q^{-1} = \frac{q}{|q|^2} \quad (2.62)$$

Donde:

- ✓ q cuaternión.
- ✓ q_0 parte escalar del cuaternión.
- ✓ q_1i, q_2j, q_3k parte vectorial del cuaternión.
- ✓ q^{-1} cuaternión inverso.

Después de la captura de datos y de seleccionar el punto de referencia, se obtiene dos vectores que alojan la posición actual y la posición anterior (referencia) guardados en

cuaterniones, a partir de estos datos se obtiene los valores de roll, pitch y yaw, de cada vector orientación para el brazalete MYO 1.

2.10.2. Interfaz MYO 2

La interfaz desarrollada para la lectura de los datos de la IMU del MYO 2, se muestra en la Figura 2.43.



Figura 2.43. Interfaz de conexión del brazalete MYO 2.
(Fuente: Propia)

En esta interfaz también se usa el comando *tcpip* de Matlab para establecer la conexión remota con el computador 1 (principal). Después de la captura de datos y de seleccionar el punto de referencia, se obtiene dos vectores que alojan la posición actual y la posición anterior (referencia) guardados en cuaterniones, a partir de estos datos se obtiene los valores de roll, pitch y yaw, de cada vector orientación para el brazalete MYO 2.

2.11. Implementación del entorno virtual

Para el desarrollo de la simulación del entorno virtual se ha empleado Simulink-Matlab 2018a y el toolkit Simscape Multibody, siendo necesario el Plug-in Simscape Multibody Link, el cual permite exportar modelos CAD de Autodesk Inventor a modelos Simscape Multibody (<https://la.mathworks.com/help/phymod/smlink/ug/installing-and-linking-simmechanics-link-software.html>).

Para poder hacer uso del modelo CAD del robot se ha empleado el comando “smimport”. Para la simulación de los movimientos se ha implementado las ecuaciones dinámicas de la Sección 1.6. Además, para probar el sistema en conjunto, se ha establecido una misión en la cual un usuario operará remotamente al robot para trasladar un objeto entre dos puntos A y B representados por círculos (verde y amarillo oscuro), tal como muestra la Figura 2.44.

Dichos puntos están ubicados dentro del espacio de trabajo del robot, y con el uso de las ecuaciones cinemáticas se detecta si el extremo del robot está sobre uno de los mismos,

para las acciones correspondientes de tomar o dejar el objeto representado por una esfera. Para notificar al usuario que el extremo del robot está ya sea en A o B, cada punto tomará un color más claro.

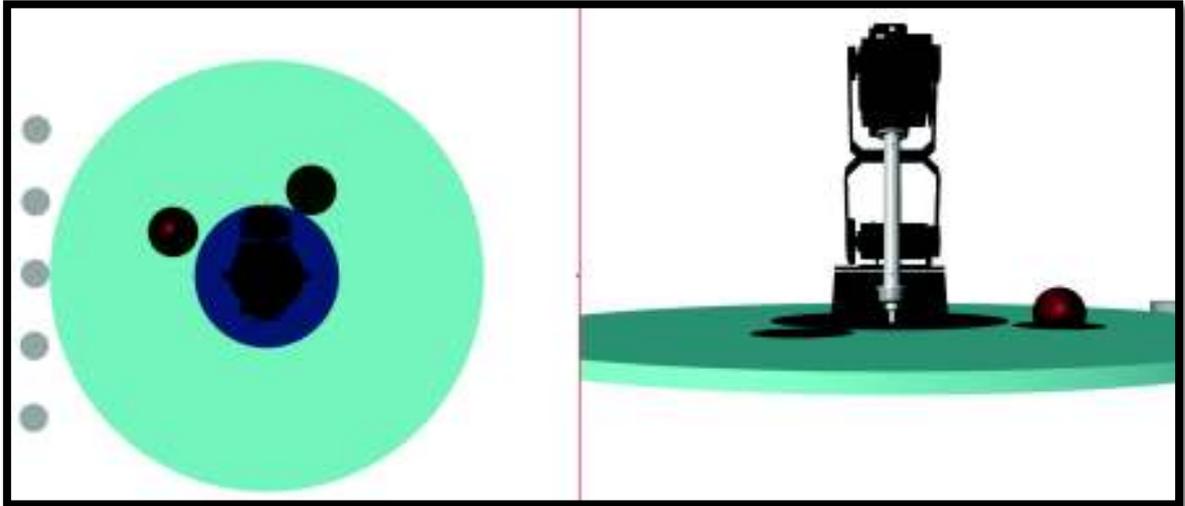


Figura 2.44. Robot manipulador en el entorno virtual.
(Fuente: Propia)

Además, en la parte izquierda de la Figura 2.44 se muestra 5 círculos de color gris, los cuales se encenderá uno a la vez, con el fin de indicar al usuario la realización de uno de los gestos, éstos se enumeran de del 1 al 5 de abajo arriba, de manera que se tiene el siguiente mapeo:

- 1: wave-out.
- 2: wave-in.
- 3: wave-close.
- 4: wave-Thumb.
- 5: wave-index.

Sino se enciende ningún círculo, significa que se está realizando el gesto de relajación. Con el fin de mostrar una aplicación en el uso de gestos, usando los gestos 1 y 2, se puede cambiar el color del objeto (rojo, verde, azul), así como su masa en 5g, 20g y una masa variable de 100 a 5g, para simular por ejemplo el derrame de un líquido de un recipiente.

2.12. Sistema de operación remota

El objetivo principal es controlar al manipulador remotamente, para lo cual se ha considerado el esquema de la Figura 2.45.



Figura 2.45. Sistema para operación remota.
(Fuente: Propia)

Sistema maestro: PC 1, es la estación de control que está constituida por la interfaz basada en los sensores MYO 1 y MYO 2 y un computador con los algoritmos de procesamiento. El sistema maestro envía una trama de datos que permite el control sobre el sistema esclavo.

Sistema esclavo: PC 2, constituye el robot virtual junto con el controlador implementado en un computador.

Sistema de comunicación: comprende un dispositivo router, con el que se genera una red inalámbrica LAN para la comunicación entre los sistemas maestro y esclavo. Es así que en sistemas tele operados, es necesario trabajar con un protocolo de comunicación que ofrezca seguridad y una buena velocidad de transmisión de datos, por lo que se ha empleado el protocolo TCP/IP para asegurar que los datos lleguen correctamente entre los dos computadores.

En este trabajo para la realimentación del sistema esclavo hacia la estación de control, se ha considerado de tipo visual, a partir de la cual se tomarán decisiones sobre el robot virtual. En este caso se consideró en primer lugar el envío de capturas de pantalla del entorno virtual a la estación de control. En este contexto se ha empleado el software vMix (vMix Desktop capture) que permite realizar un streaming de la pantalla, para lo cual se ha empleado la misma red de comunicación mencionada (Figura 2.46).

En segundo lugar, debido al retardo existente en el streaming, se ha buscado otra opción mediante el uso de un teléfono celular como una cámara IP (aplicación Android IP-CAM) para transmisión de video del entorno virtual enfocado tal como se muestra en la Figura 2.47.

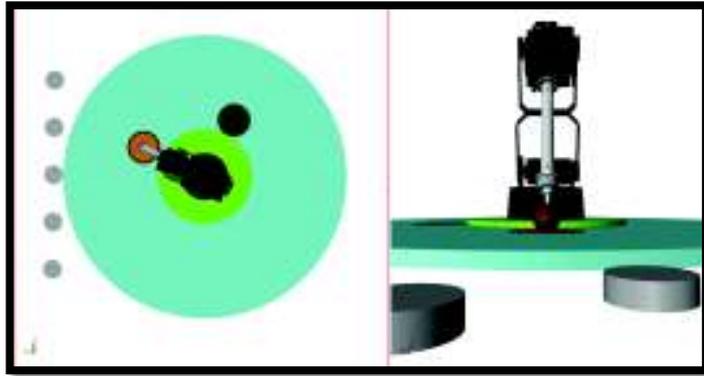


Figura 2.46. Realimentación visual vMix.
(Fuente: Propia)



Figura 2.47. Sistema de realimentación con cámara IP (Celular).
(Fuente: Propia)

Finalmente, se utilizó la conexión remota usando 2 celulares para manejar remotamente el manipulador virtual. Ya que como se mencionó el retardo en el streaming era menor, con lo que los movimientos son más fluidos.

3. PRUEBAS Y RESULTADOS

En este capítulo se expondrán las pruebas realizadas de los sistemas y algoritmos implementados en este proyecto.

3.1. Pruebas off-line del sistema robótico virtual (Esclavo)

No hay un criterio estándar para evaluar el desempeño de controladores. En algunos casos de sistemas no lineales, la medición del desempeño se basa en la revisión de las gráficas del error de posición, siendo un método subjetivo. De manera que en este trabajo se ha hecho uso de los índices: Error Cuadrático Integrado (ISE) y el Error Absoluto Integrado (IAE), de los cuales un valor bajo indicará un buen desempeño del sistema de control.

$$ISE = \int e^2 dt; \quad IAE = \int |e| dt \quad (3.1)$$

En el caso del controlador CIA, fue necesario hacer simulaciones con el fin de determinar un valor adecuado para los elementos de la diagonal de la matriz Γ que influye en la ley de adaptación de la Ecuación 2.39. Ya que como en el caso del controlador CPC, se espera que haya un desacoplamiento en lazo cerrado, se consideró que los elementos de la diagonal son iguales. En este punto se realizaron simulaciones en el cual se pudo evidenciar que un valor pequeño de Γ implica una estimación de masas atenuada y por lo tanto una respuesta lenta del controlador (Figura 3.1 a), mientras que un valor grande Γ ocasiona ciertos impulsos e incluso oscilaciones en la señal de control evidenciando una estimación de masas brusca (Figura 3.1 b). Por lo que para valor en que se obtuvieron buenos resultados son los siguientes:

$$\Gamma = \begin{bmatrix} 20 & 0 & 0 \\ 0 & 20 & 0 \\ 0 & 0 & 20 \end{bmatrix}$$

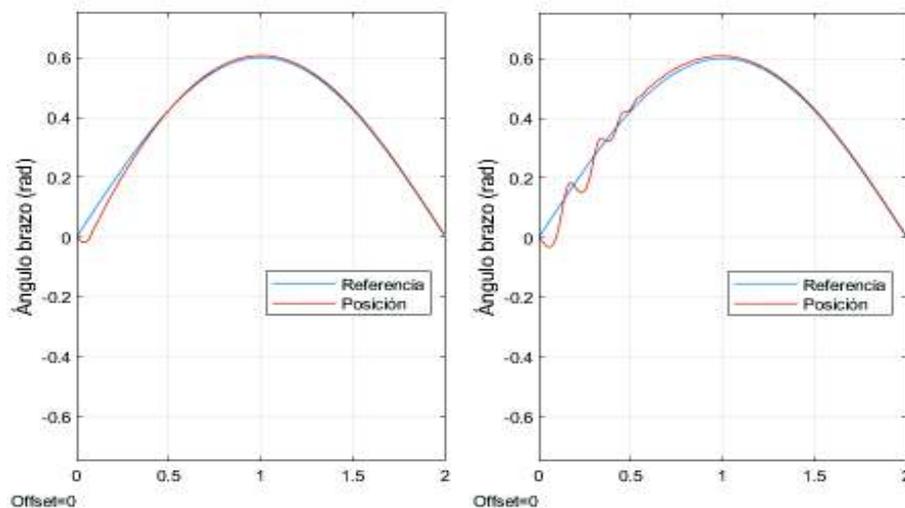


Figura 3.1. a) Respuesta para diagonal de Γ con 5; b) Respuesta para diagonal de Γ con 100. (Fuente: Propia)

Para esta simulación se consideró una señal sinusoidal la cual se describe más adelante, el tiempo de simulación considerado es de 2 segundos, y las gráficas corresponden a la respuesta en la articulación 2, resultados similares se tienen en las otras articulaciones.

A continuación, en base a los valores seleccionados de las constantes de control de cada uno de los controladores, se presentan las pruebas respectivas, para verificar el desempeño de los controladores CPC y CIA.

3.1.1. Prueba de simulación con referencia tipo escalón para CPC y CIA

En esta prueba se verifica la respuesta de los controladores ante una entrada escalón suavizada en sus bordes, esto debido a que se necesita una señal que sea continua hasta su segunda derivada, ya que en el esquema de control para CPC y CIA se requiere referencia de velocidad y aceleración angular. En este caso se usa una señal con amplitud de $\frac{\pi}{2}$ rad para cada articulación.

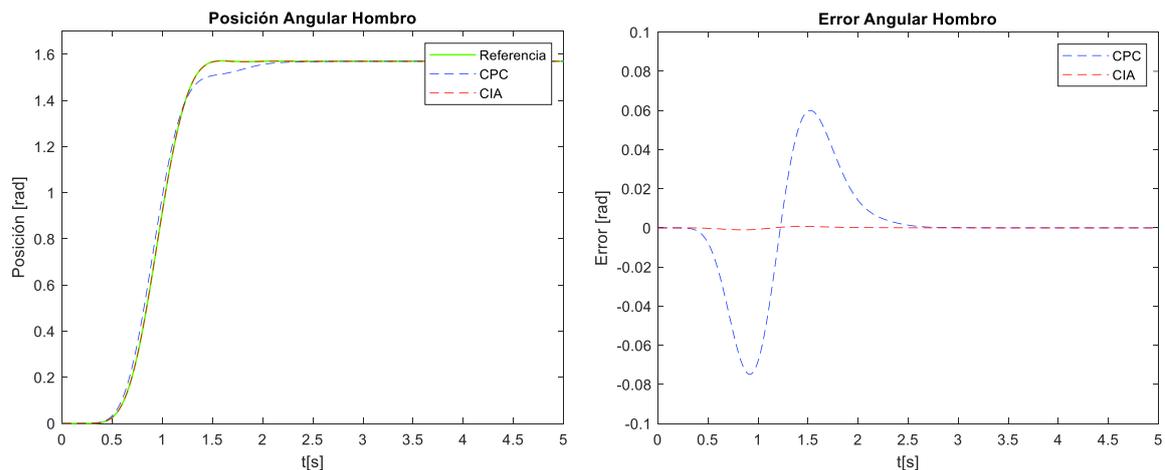


Figura 3.2. Seguimiento y error de posición angular de la articulación del hombro (q_1) - CPC y CIA. (Fuente: Propia)

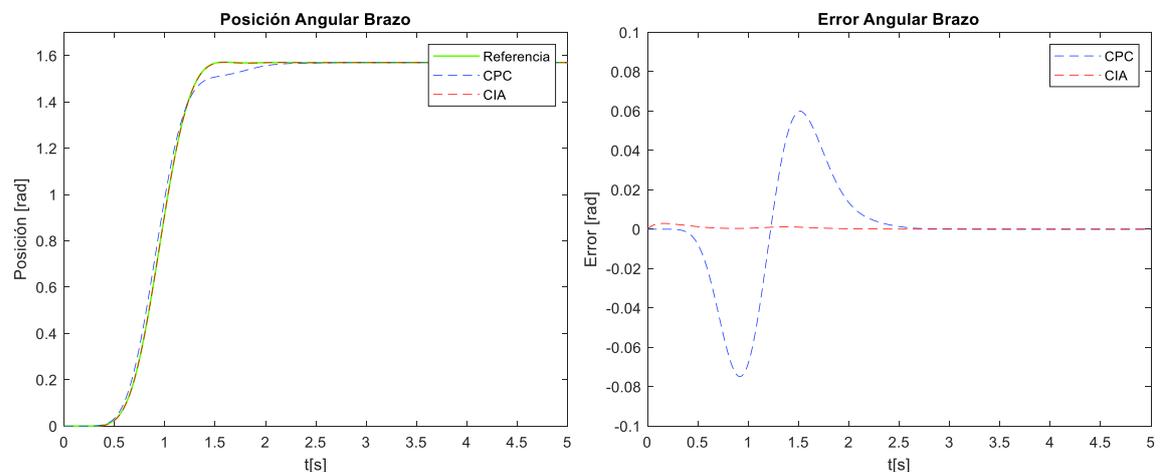


Figura 3.3. Seguimiento y error de posición angular de la articulación del brazo (q_2) - CPC y CIA. (Fuente: Propia)

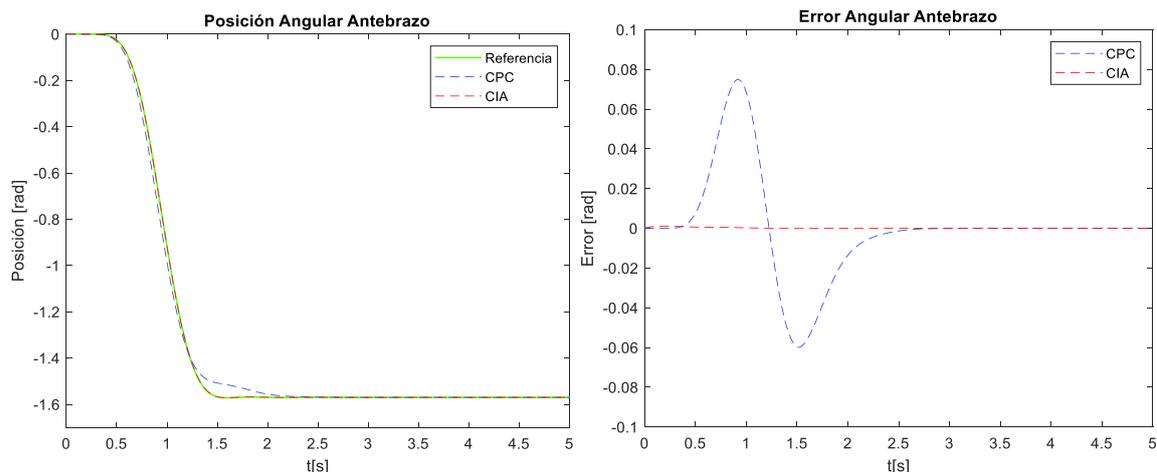


Figura 3.4. Seguimiento y error de posición angular de la articulación del antebrazo (q_3) - CPC y CIA.

(Fuente: Propia)

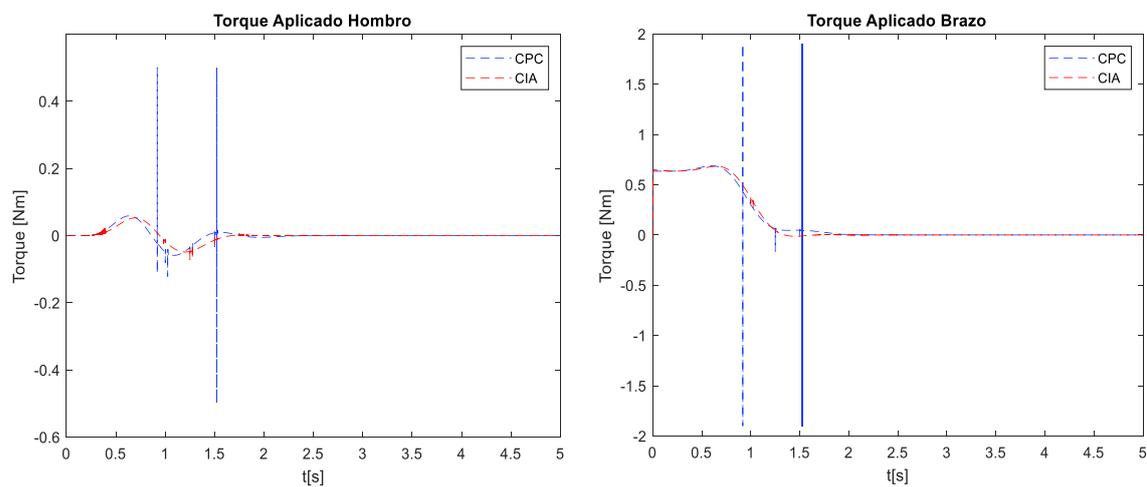


Figura 3.5. Torque aplicado a las articulaciones del hombro y brazo (q_1, q_2) – CPC y CIA.

(Fuente: Propia)

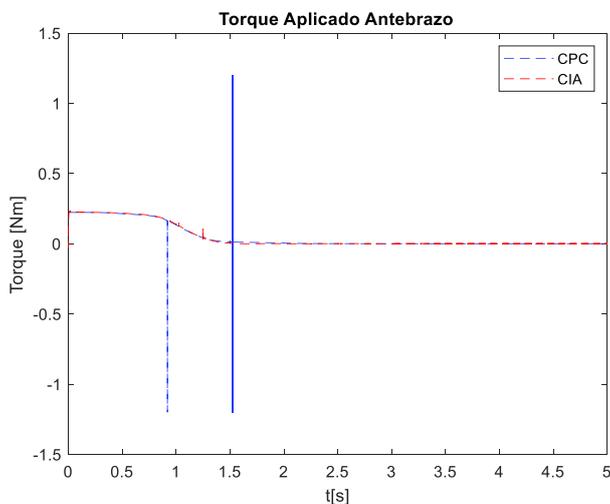


Figura 3.6. Torque aplicado a la articulación del antebrazo (q_3) – CPC y CIA.

(Fuente: Propia)

En las gráficas de posición angular se puede verificar que el tiempo de establecimiento es menor a 1[s], cumpliendo con el parámetro de diseño de t_s . Así mismo, en la Tabla 3.1 se muestran los valores bajos de los índices ISE e IAE en los controladores CPC y CIA, concluyendo que ambos tienen un buen desempeño.

Tabla 3.1. Desempeño de controladores mediante IAE e ISE.

Controlador	Articulación	IAE	ISE
CPC	q_1	6.61×10^{-2}	3.24×10^{-3}
	q_2	6.61×10^{-2}	3.24×10^{-3}
	q_3	6.61×10^{-2}	3.24×10^{-3}
CIA	q_1	8.8×10^{-4}	5.49×10^{-7}
	q_2	2.2×10^{-3}	3.46×10^{-6}
	q_3	7.6×10^{-4}	4.87×10^{-7}

(Fuente: Propia)

En los resultados presentados previamente, se han considerado las 3 articulaciones del robot, en las pruebas que se presentarán a continuación estarán enfocadas en una sola articulación, ya que en las otras se obtienen resultados similares. En este contexto se presentarán los resultados obtenidos para la articulación 2, ya que estructuralmente es la que soporta en mayor parte el peso de los eslabones (brazo y antebrazo).

3.1.2. Prueba de simulación con referencia sinusoidal para CPC y CIA

La respuesta de los controladores se verifica aplicando a las articulaciones la trayectoria

$$q = \left[0.3 \sin\left(\frac{\pi}{2}t\right) \quad 0.6 \sin\left(\frac{\pi}{2}t\right) \quad 0.5 \cos\left(\frac{\pi}{2}t\right) \right], \text{ para lo cual se ha considerado 10 segundos de simulación.}$$

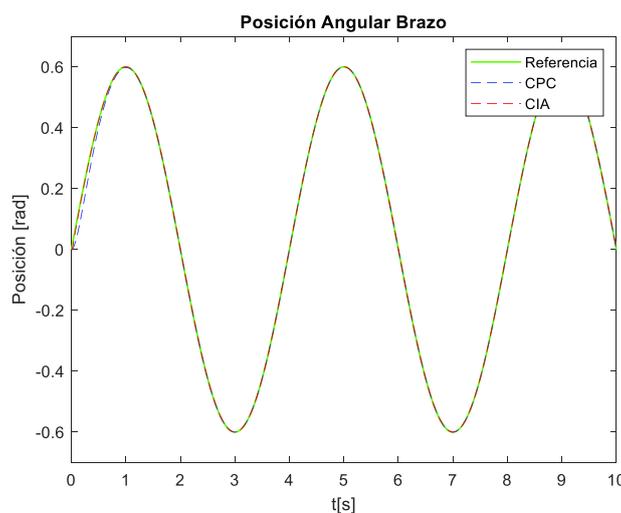


Figura 3.7. Posición angular de la articulación del brazo (q_2) para CPC y CIA.
(Fuente: Propia)

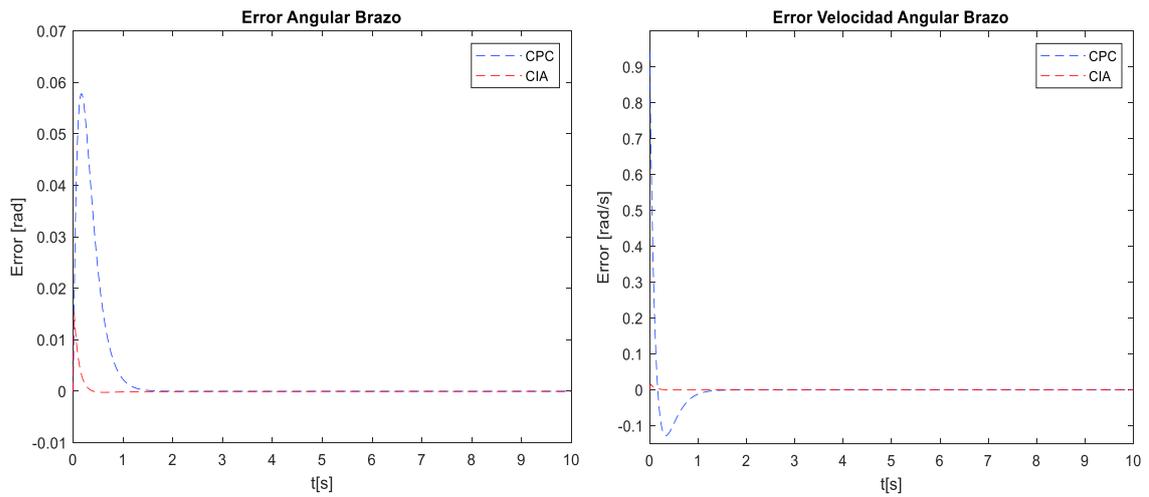


Figura 3.8. Errores de posición y velocidad de la articulación del brazo (q_2) para CPC y CIA. (Fuente: Propia)

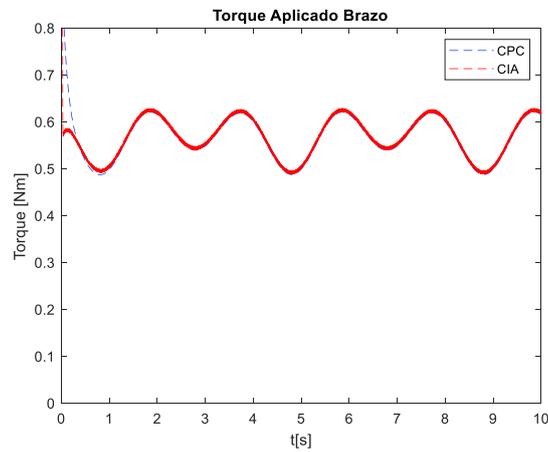


Figura 3.9. Torque aplicado a la articulación del brazo (q_2) para CPC y CIA. (Fuente: Propia)

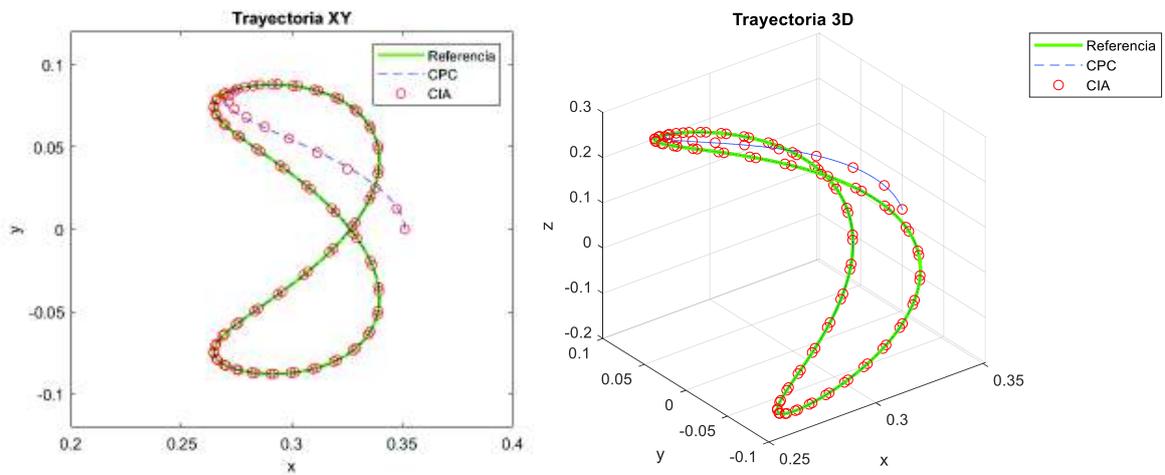


Figura 3.10. Trayectoria efectuada por el extremo del robot en el plano XY y en el espacio 3D. (Fuente: Propia)

En la Tabla 3.2 se muestran los índices ISE e IAE de los controladores, observándose que ambos tienen un buen desempeño durante toda la trayectoria, siendo aún mejor en CIA.

Tabla 3.2. Desempeño de controladores mediante IAE e ISE.

Controlador	Articulación	IAE	ISE
CPC	q_1	1.31×10^{-2}	2.57×10^{-4}
	q_2	2.62×10^{-2}	1.03×10^{-3}
	q_3	1.67×10^{-1}	5.21×10^{-2}
CIA	q_1	1.34×10^{-3}	2.55×10^{-6}
	q_2	2.04×10^{-3}	1.37×10^{-5}
	q_3	1.67×10^{-1}	4.16×10^{-2}

(Fuente: Propia)

3.1.3. Prueba de simulación con referencia sinusoidal y variación de masa

En este trabajo no se ha considerado como objetivo estudiar la dinámica de los actuadores, pues sería necesario disponer de los mismos para extraer los modelos y añadirlos a la Ecuación 1.40 de la dinámica del manipulador. Por otro lado, la estructura robótica puede emplear servomotores, los cuales en general contienen un sistema de control interno, siendo dispositivos generadores de torque. En este contexto y en base a [6,16], se ha considerado a los actuadores como un sistema primer orden, siendo para el caso usado como una perturbación a los controladores, dada por. $G(s) = \frac{50}{s+50}$

Además, dentro de aplicaciones reales, el uso de manipuladores está ligado al manejo de herramientas o sujeción de objetos, siendo esto un problema para los sistemas de control, ya que se altera la dinámica del sistema. En base a lo mencionado, y con el fin de verificar la robustez del controlador CIA con respecto al CPC, se considera la sujeción de diferentes masas: 5g, 20g y variable de 100 a 5g. El tiempo de simulación en estas pruebas es de 10 segundos.

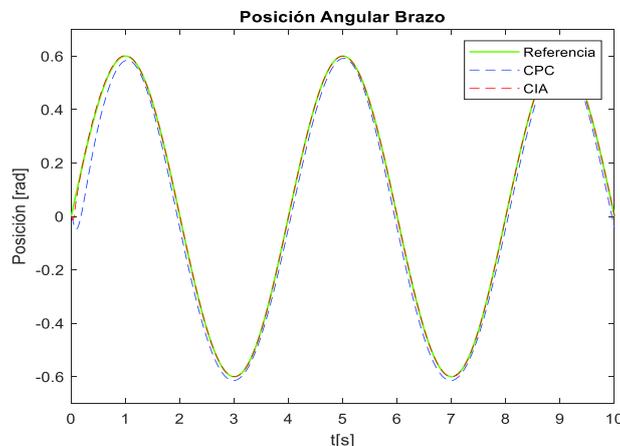


Figura 3.11. Evolución temporal de la articulación del brazo (q_2) para CPC y CIA (masa de objeto 5g).

(Fuente: Propia)

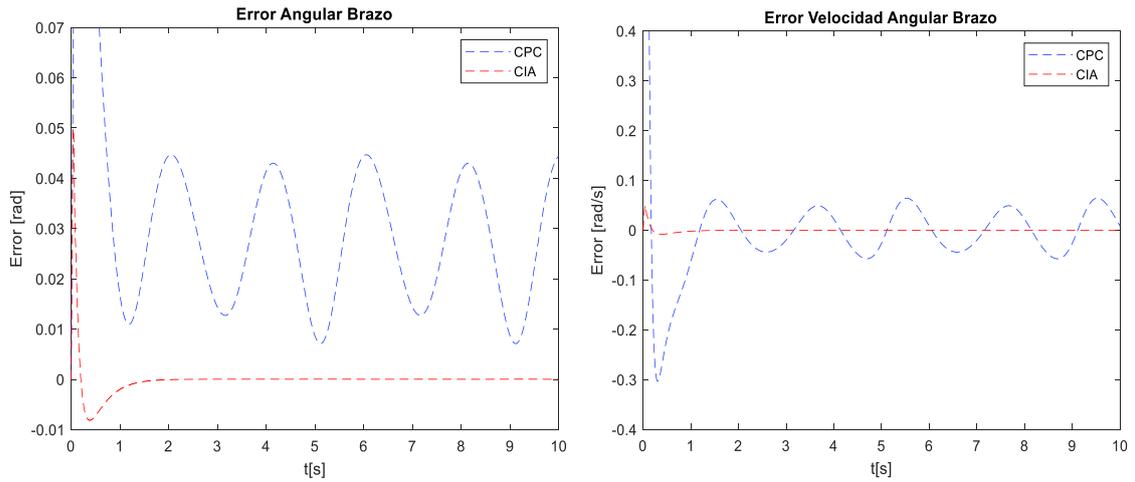


Figura 3.12. Errores de posición y velocidad de la articulación del brazo (q_2) para CPC y CIA (masa de objeto 5g).
(Fuente: Propia)

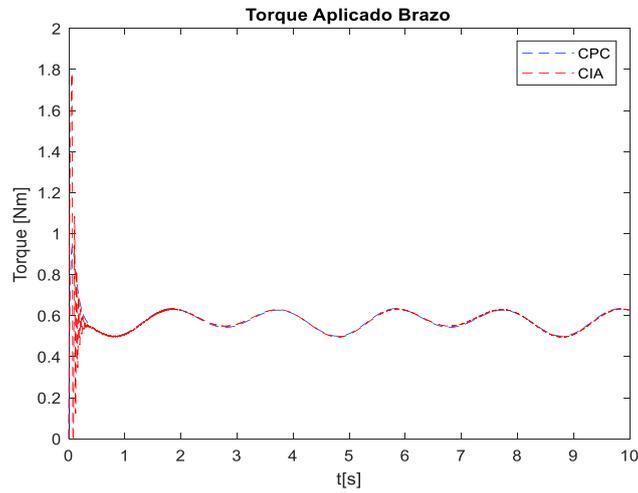


Figura 3.13. Torque aplicado a la articulación del brazo (q_2) para CPC y CIA (masa de objeto 5g).
(Fuente: Propia)

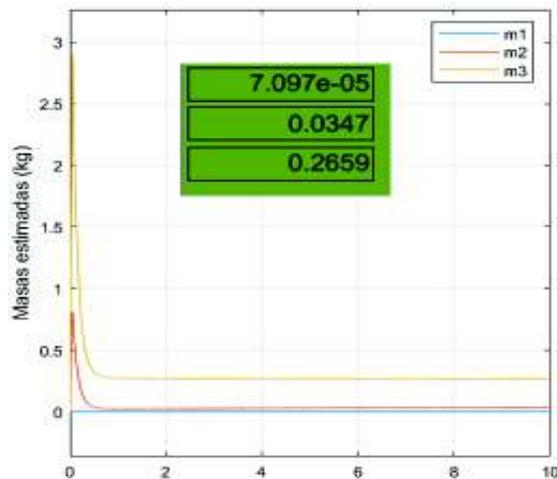


Figura 3.14. Estimación de masas de los eslabones en el controlador CIA por medio de la Ecuación 2.42 (masa de objeto 5g).
(Fuente: Propia)

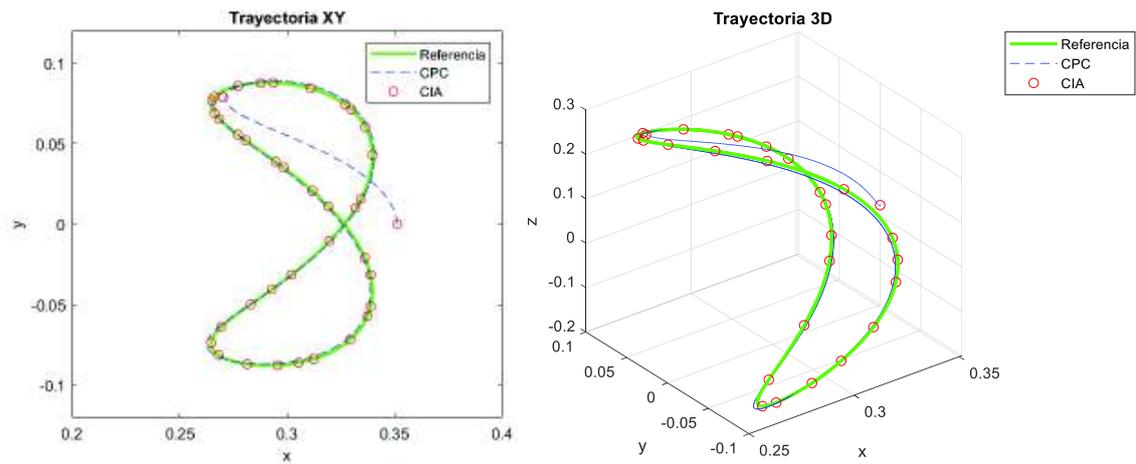


Figura 3.15. Trayectoria efectuada por el extremo del robot en el plano XY y en el espacio 3D (masa de objeto 5g).
(Fuente: Propia)

En la Figura 3.14 se puede observar una buena estimación de las masas, bajo condiciones iniciales de masa cero. En la Tabla 3.3 se muestran los índices ISE e IAE de los controladores, observándose que ambos tienen un buen desempeño durante toda la trayectoria, siendo mejor el CIA.

Tabla 3.3. Desempeño de controladores mediante IAE e ISE.

Controlador	Articulación	IAE	ISE
CPC	q_1	1.71×10^{-2}	2.7×10^{-4}
	q_2	3.27×10^{-1}	1.73×10^{-2}
	q_3	3.62×10^{-1}	4.31×10^{-2}
CIA	q_1	2.99×10^{-3}	1.23×10^{-5}
	q_2	9.32×10^{-3}	1.69×10^{-4}
	q_3	1.65×10^{-1}	4.16×10^{-2}

(Fuente: Propia)

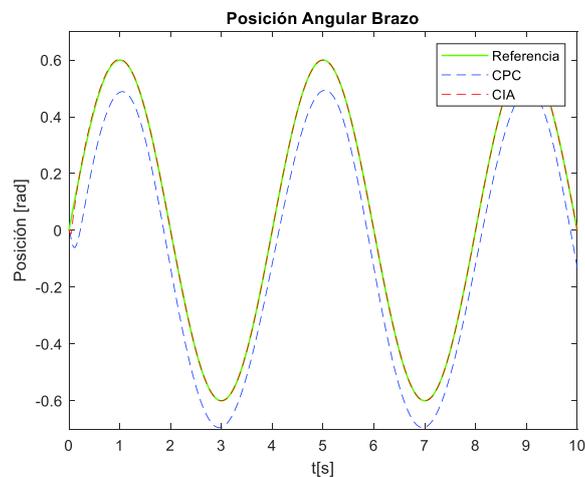


Figura 3.16. Evolución temporal de la articulación del brazo (q_2) para CPC y CIA (masa objeto 10g).
(Fuente: Propia)

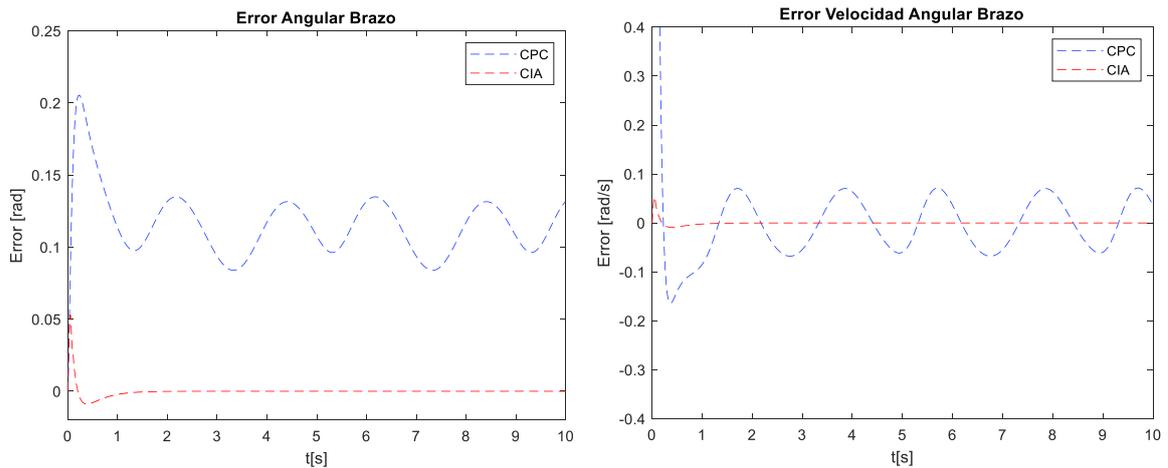


Figura 3.17. Errores de posición y velocidad de la articulación del brazo (q_2) para CPC y CIA (masa de objeto 10g).
(Fuente: Propia)

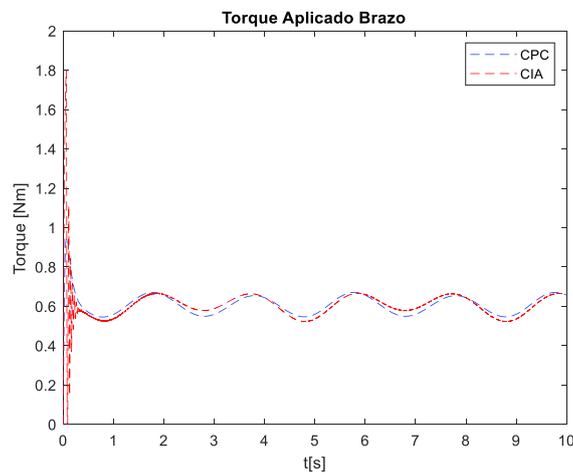


Figura 3.18. Torque aplicado a la articulación del brazo (q_2) para CPC y CIA (masa de objeto 10g).
(Fuente: Propia)

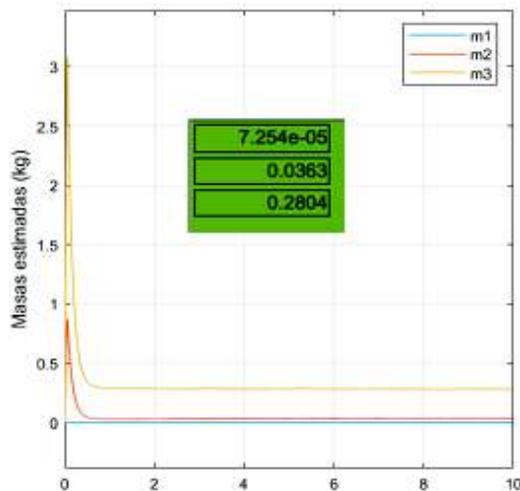


Figura 3.19. Estimación de masas de los eslabones en el controlador CIA por medio de la Ecuación 2.42 (masa de objeto 10g).
(Fuente: Propia)

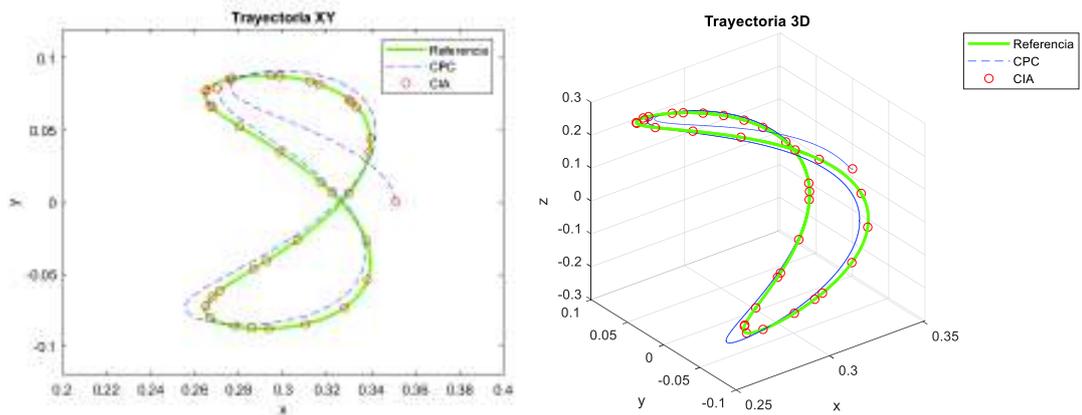


Figura 3.20. Trayectoria efectuada por el extremo del robot en el plano XY y el espacio 3D (masa objeto 10g).
(Fuente: Propia)

En la Figura 3.19 se muestra nuevamente una buena estimación de las masas, con condiciones iniciales de masa cero. En la Tabla 3.4 se muestran los índices ISE e IAE de los controladores, notándose que el desempeño del CPC disminuye, mientras que el de CIA es mejor, siendo evidente en la Figura 3.20 donde la trayectoria de CPC se desvía de la referencia.

Tabla 3.4. Desempeño de controladores mediante IAE e ISE.

Controlador	Articulación	IAE	ISE
CPC	q_1	2.28×10^{-2}	3.09×10^{-4}
	q_2	1.16×10^0	1.39×10^{-1}
	q_3	1.01×10^0	1.28×10^{-1}
CIA	q_1	3.12×10^{-3}	1.35×10^{-5}
	q_2	1.01×10^{-2}	1.9×10^{-4}
	q_3	1.64×10^{-1}	4.15×10^{-2}

(Fuente: Propia)

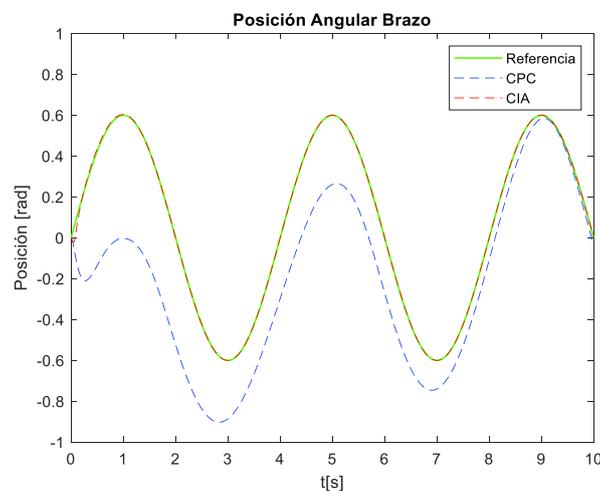


Figura 3.21. Evolución temporal de la articulación del brazo (q_2) para CPC y CIA (masa objeto 100-5g).
(Fuente: Propia)

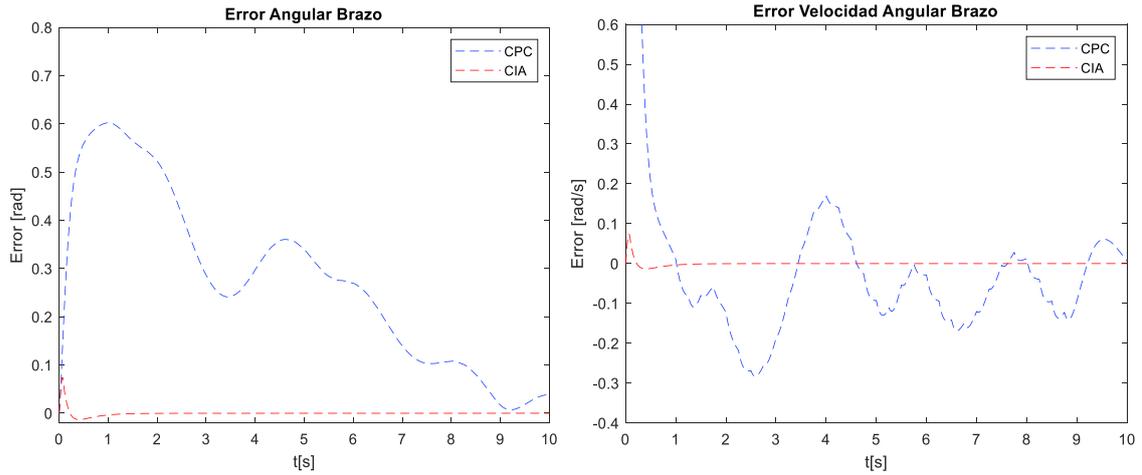


Figura 3.22. Errores de posición y velocidad de la articulación del brazo (q_2) para CPC y CIA (masa de objeto 100-5g).
(Fuente: Propia)

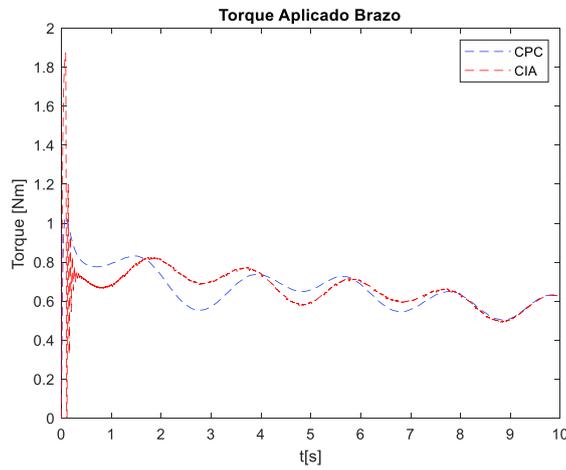


Figura 3.23. Torque aplicado a la articulación del brazo (q_2) para CPC y CIA (masa de objeto 100-5g).
(Fuente: Propia)

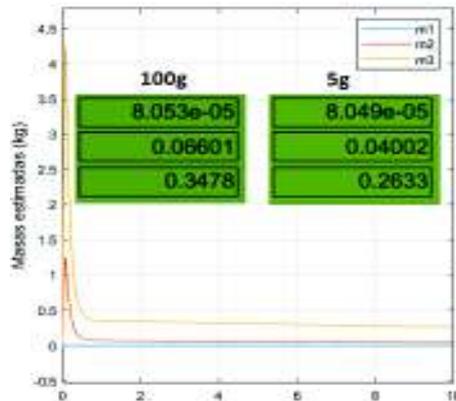


Figura 3.24. Estimación de masas de los eslabones en el controlador CIA por medio de la Ecuación 2.42 (masa de objeto 100-5g).
(Fuente: Propia)

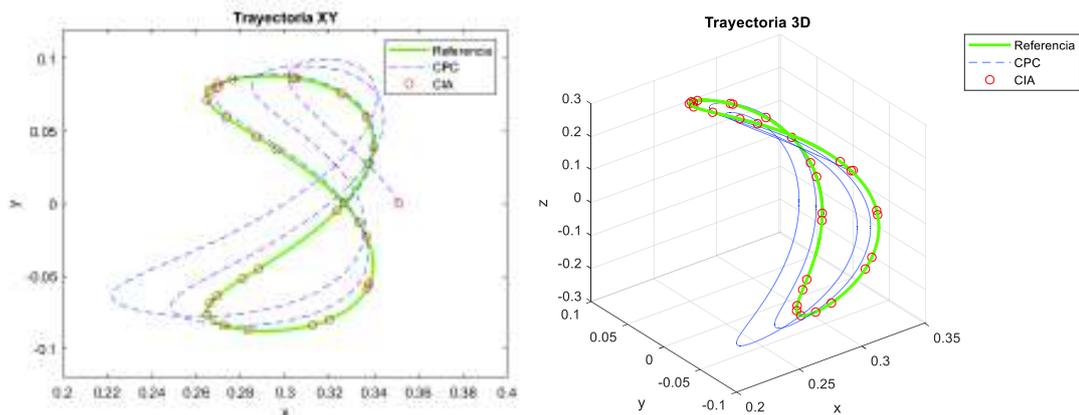


Figura 3.25. Trayectoria efectuada por el extremo del robot en el plano XY y espacio 3D (masa de objeto 100-5g).
(Fuente: Propia)

En la Figura 3.24 también se muestra una buena estimación de las masas, con condiciones iniciales de masa cero. En la Tabla 3.5 se muestran los índices ISE e IAE de los controladores, observándose que el desempeño del CPC disminuye aún más, mientras que el de CIA se mantiene con un buen desempeño, siendo evidente en la Figura 3.25 donde la trayectoria de CPC en un al inicio tiene una gran desviación de la referencia.

Tabla 3.5. Desempeño de controladores mediante IAE e ISE.

Controlador	Articulación	IAE	ISE
CPC	q_1	3.91×10^{-2}	5.34×10^{-4}
	q_2	2.77×10^0	1.09×10^0
	q_3	2.17×10^0	7.21×10^{-1}
CIA	q_1	3.85×10^{-3}	1.99×10^{-5}
	q_2	1.64×10^{-2}	4.71×10^{-4}
	q_3	1.6×10^{-1}	4.02×10^{-2}

(Fuente: Propia)

En base a las gráficas de posición angular se aprecia que en ambos controladores se logra una respuesta críticamente amortiguada, con un tiempo de respuesta dentro del tiempo estimado para la determinación de las constantes de los controladores.

En las gráficas de los torques aplicados, se puede observar que los mismos al estar acotados, los picos son limitados evitando afectar a la dinámica del sistema.

Como se ha visto la variación de un parámetro en este caso la masa del extremo afecta a la dinámica del sistema, es por eso que el CPC disminuye su desempeño, aún más si la variación de este parámetro es mayor. En el caso del CIA en base al error, trata de adaptar la dinámica simulada para disminuir el error.

3.2. Pruebas off-line de la interfaz de control (maestro)

Las pruebas para evaluar tanto el sistema en conjunto como para evaluar la interfaz de control han sido realizadas en 20 usuarios nuevos, ninguno perteneciente al grupo de entrenamiento y con edades comprendidas entre 18-30 años y que no habían interactuado antes con el sistema de reconocimiento de gestos (Figura 3.26). Con el fin de acoger la opinión de los usuarios en la prueba off-line y on-line de la interfaz de control se ha planteado una encuesta que se presenta en el Anexo1.

En este apartado como primero punto se presentará los resultados que permiten valorar el sistema de reconocimiento de los 6 gestos establecidos. Para esta evaluación se pidió a cada usuario que interactúe con el sistema durante 5 minutos para que se familiarice con los 6 gestos que debe realizar. Cabe mencionar que cada gesto se ligó a diferentes movimientos del robot virtual sin el control para compensar la variación de la masa del objeto seleccionado, esto con el objetivo de hacer la prueba más interactiva y que el usuario no lo vea tedioso.



Figura 3.26. Prueba del sistema de reconocimiento de gestos.
(Fuente: Propia)

Transcurrido el tiempo de familiarización se pide a cada usuario realice 10 repeticiones de cada gesto, los aciertos y errores en el reconocimiento de gestos por el sistema de reconocimiento implementado en este proyecto son registrados.

A continuación, en la Tabla 3.6 se presenta los resultados del reconocimiento de los 20 nuevos usuarios, observándose que el sistema de reconocimiento de gestos tiene un acierto del 91.6% para las pruebas realizadas.

Tabla 3.6. Resultados de reconocimiento de 20 usuarios.

	OUT	IN	CLOSE	INDEX	THUMB	RELAX	TOTAL
Aciertos	200	178	198	167	156	200	
Porcentaje de acierto	100	89	99	83.5	78	100	91.6
No aciertos	0	22	2	33	44	0	
Porcentaje de no aciertos	0	11	1	16.5	22	0	8.4

(Fuente: Propia)

Se puede observar que el menor porcentaje de acierto se produce en el gesto thumb. Esto se debe en gran medida a que los nuevos usuarios no tienen mucha práctica al momento de realizar el gesto y pierden intensidad al momento de realizar una nueva repetición.

Después de esta primera fase de pruebas, se le se le pide al usuario que responda a las 6 primeras preguntas de la encuesta:

1. Según su criterio en el test, ¿qué tan bien el sistema reconoce el gesto Wave-out?
2. Según su criterio en el test, ¿qué tan bien el sistema reconoce el gesto Wave-in?
3. Según su criterio en el test, ¿qué tan bien el sistema reconoce el gesto Close?
4. Según su criterio en el test, ¿qué tan bien el sistema reconoce el gesto Thumb?
5. Según su criterio en el test, ¿qué tan bien el sistema reconoce el gesto Relax?
6. Según su criterio en el test, ¿qué tan bien el sistema reconoce el gesto Index?

Cada pregunta tiene las siguientes opciones de respuesta:

- | | |
|------------------|--|
| <i>Excelente</i> | |
| <i>Bueno</i> | |
| <i>Regular</i> | |
| <i>Malo</i> | |

Para obtener un valor cuantitativo de la encuesta se asignó una ponderación en porcentaje a cada opción de respuesta:

- | | |
|------------------|------|
| <i>Excelente</i> | 100% |
| <i>Bueno</i> | 75% |
| <i>Regular</i> | 50% |
| <i>Malo</i> | 25% |

Las primeras 6 preguntas de la encuesta muestran una buena aceptación de los usuarios cuando realizan cada uno de los gestos propuestos.

Tabla 3.7. Valoración del reconocimiento de gestos según los usuarios

	Out	In	Close	Index	Thumb	Relax
Promedio [%]	93.7500	90.0000	92.5000	81.2500	91.2500	83.7500

(Fuente: Propia)

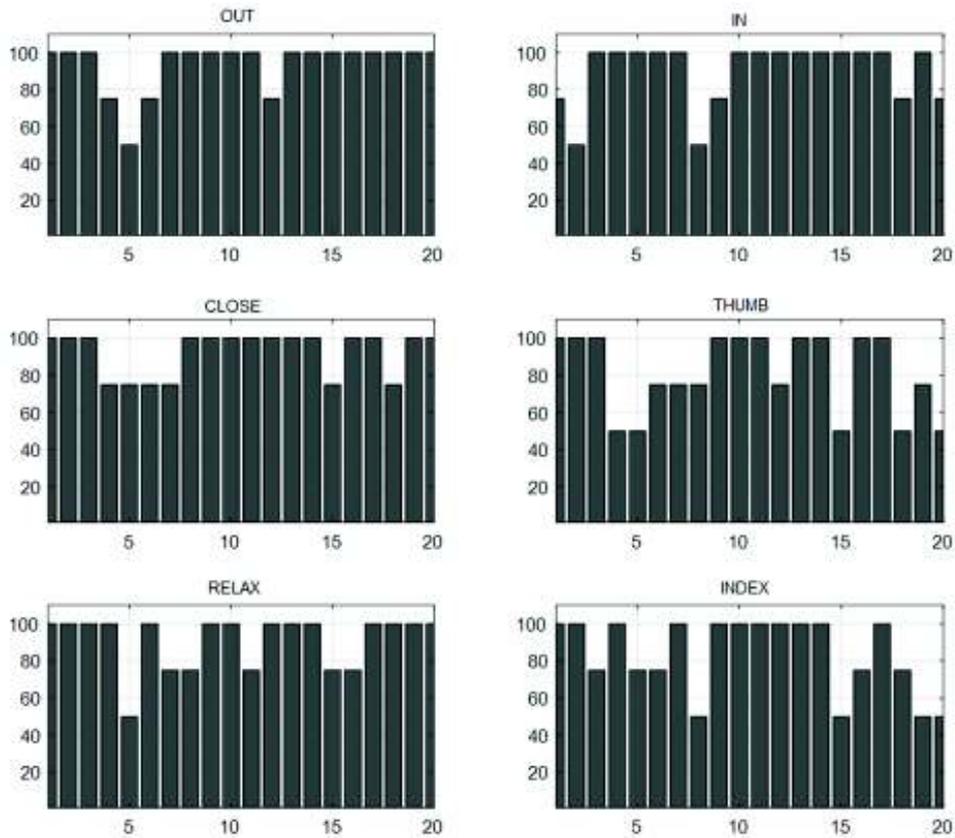


Figura 3.27. Valoración 20 usuarios, gestos Out, Close, Relax, In, Thumb, Index. (Fuente: Propia)

3.3. Pruebas on-line para operación remota del robot virtual

En este ítem se presenta las pruebas realizadas del sistema en conjunto, es decir la interfaz de control y el sistema esclavo (robot virtual) para los 20 usuarios, para lo cual el escenario es tal que ambos sistemas se encuentran conectados a una misma red de área local inalámbrica.

En ese aspecto se describirá la metodología y las consideraciones al realizar las pruebas de funcionamiento del sistema y el desempeño del operador con el sistema en conjunto.

3.3.1. Consideraciones

A continuación, se expone el entorno y condiciones en las cuales se llevaron a cabo las pruebas para operación remota del sistema, permitiendo usar el reconocimiento de gestos para interactuar con el robot, y el desempeño del operador en el control del movimiento del robot virtual. El sitio dónde se ha realizado las pruebas es en el laboratorio de Inteligencia y Visión Artificial de la Facultad de Ingeniería en Sistemas-EPN.

Para el control remoto se ha empleado una red inalámbrica de área local dedicada, para lo cual se ha empleado un Router D-Link Modelo DIR615, el cual asigna una dirección IP

tanto al computador donde se conecta la interfaz de control, como al computador donde se encuentra el robot virtual. El tráfico que circula por este router es únicamente el de las conexiones entre las dos PC.

Para el sistema de realimentación, se ha probado el streaming usando el software Vmix, observándose que el delay dificulta la operación del robot virtual, por lo que las pruebas en su mayoría se ha empleado el uso de un teléfono celular como cámara IP mediante la aplicación Android IP-CAM, véase la Figura 2.47.

Para la operación remota se ha considerado distancias en un promedio de 5 metros entre el esclavo y la interfaz de control, esto con el fin de indicar a los usuarios la aplicación potencial a futuro del sistema como un sistema tele-operado. Dentro del entorno virtual se establecieron dos puntos A y B de los cuales, el usuario debe tomar un objeto de un punto del espacio de trabajo del robot para ubicarlo en otro punto. Debido a la dificultad causada por el objeto al controlador CPC para seguir la referencia de trayectoria, las pruebas se realizaron directamente con el CIA. Por defecto el objeto aparece el punto A, de manera que el usuario pueda trasladarlo al punto B (Figura 3.28).

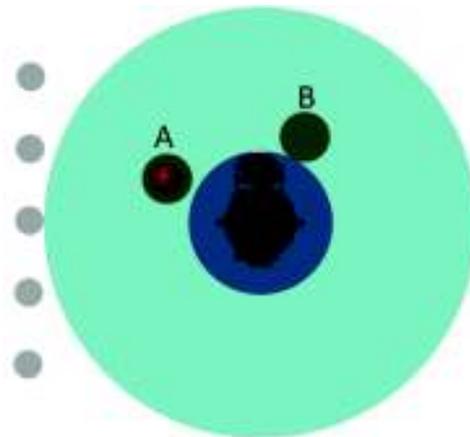


Figura 3.28. Puntos establecidos en el entorno virtual.
(Fuente: Propia)

De acuerdo a la Figura 2.44 se ha considerado dos vistas de realimentación del robot al operador, una que es una vista superior con la cual el usuario podrá girar la base y extender el extremo para ubicar sobre un punto del espacio; una vista frontal mediante una cámara móvil (virtual) que enfoca al extremo del robot para subir o bajar el mismo, de acuerdo a los movimientos explicados en la Sección 2.2.

En base al tipo de controlador seleccionado el operador sentirá una menor dificultad para operar al sistema, lo cual se verá reflejado en los tiempos en que el usuario puede mover al robot entre los puntos objetivos.

3.3.2. Metodología

Con el fin de realizar la prueba final con cada uno de los usuarios y recopilar información acerca de la maniobrabilidad, se realizó el siguiente procedimiento:

- Indicar al usuario la ubicación en la que deben colocarse los brazaletes en el brazo y antebrazo de acuerdo a la Figura 2.3.
- Indicar al usuario se realice la calibración de los brazaletes e ingreso al sistema de interfaz de control.
- Indicar al usuario que mediante los gestos wave-in y wave-out puede cambiar el valor de la masa del objeto, así como el color de dicho objeto y con el gesto Thumb se almacenan los valores de masa y color seleccionados.
- Establecer la conexión de los computadores mediante TCP/IP Matlab, verificando las direcciones IP de los computadores.
- Familiarizar al usuario con el entorno virtual, y movimientos que debe realizar para mover al robot.
- Dar indicación al usuario para que pueda conectarse o desconectarse con el robot virtual a través del gesto thumb.
- Familiarizar al usuario con la realización de movimientos para controlar al robot y calibrar los sensores IMU de los brazaletes en caso de ser necesario.
- Detallar la misión que debe realizar el usuario para sujetar el objeto (close) y dejarlo en el punto indicado (index/wave-out).
- Medir el tiempo desde que sujeta el objeto desde el punto A hasta dejarlo en el B.
- Medir el tiempo desde que sujeta el objeto desde el punto B hasta dejarlo en el A.
- Llenado de la encuesta por parte del usuario.

Mediante esta prueba lo que se busca es verificar la interacción del usuario con el sistema en conjunto, para lo cual como se ha mencionado se ha considerado el tiempo que tarda el usuario de mover el extremo del robot del punto A al B.



Figura 3.29. Prueba de operación remota.
(Fuente: Propia)

Así mismo, se busca ver desde la perspectiva del usuario qué tan maniobrable es el sistema, tomando en cuenta el uso de los brazaletes y su interfaz, entorno virtual, realimentación visual, y control de movimientos del robot.

3.3.3. Resultados

En la Tabla 3.8 se muestra los tiempos de maniobra de cada usuario para los dos ejercicios realizados. Como se puede observar, estos tiempos son variables y en la mayoría disminuyen conforme el usuario se entrena en el uso del sistema. Por esa razón que los tiempos de la segunda columna son menores que la primera.

Tabla 3.8. Tiempo de demora en trasladar el objeto entre los puntos A y B

	A---->B	B---->A
Usuario	t1 [s]	t2 [s]
1	21,88	15,95
2	33,65	13,23
3	57,94	18,07
4	43,17	20,75
5	35,13	29,15
6	10,21	30,69
7	11,02	8,27
8	27,49	18,84
9	13,47	x
10	14,1	14,88
11	29,13	31,15
12	16,45	12,62
13	9,15	13,62
14	x	x
15	x	x
16	42,2	23,19
17	x	x
18	17,13	22,38
19	11,2	12,1
20	16,01	14,5
Víctor	11,7	10,3
Alex	8,815	10,27

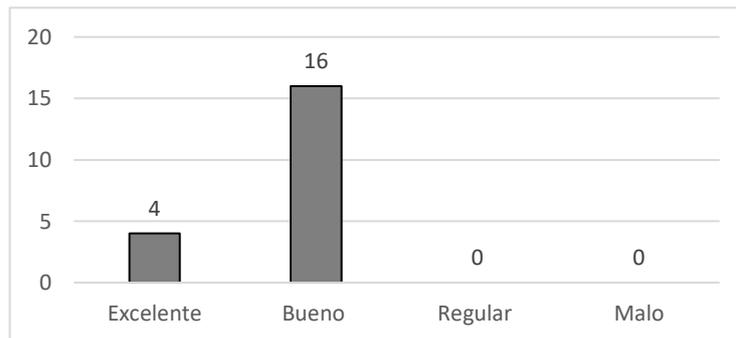
(Fuente: Propia)

Así mismo, un factor importante es el retardo en la realimentación visual ya que desfasa los movimientos del robot con respecto a los realizados por el usuario dificultando su control. El valor x indica que el usuario no pudo completar la tarea, debido a que durante las pruebas se desconectaba el brazaletes MYO 2. Este problema es debido a que al permanecer el brazaletes durante un tiempo sin censar movimiento entra en estado de ahorro de energía y se desconecta.

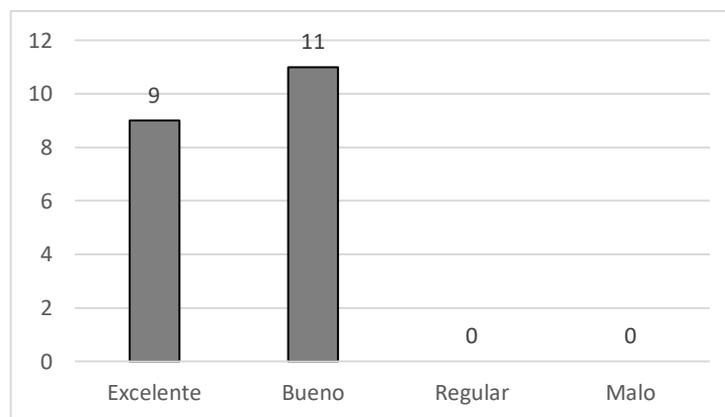
Los tiempos presentados no son comparables entre personas, ya que, dependiendo del usuario, unos pueden acoplarse al sistema más rápido que otros, por ejemplo, por algún tiempo de experiencia en entornos virtuales.

En cuanto a la maniobrabilidad del sistema se pudo verificar que depende de dos factores, uno de estos factores es la realimentación visual, pues como se mencionó este puede dificultar el control del robot sobre todo en personas que recién inician interactuando con el robot. Otro factor es el entrenamiento y práctica del usuario con el sistema sobre todo en el reconocimiento de gestos, ya que durante la operación el usuario por lo general intencionalmente realiza gestos. En este aspecto se ha considerado ciertas preguntas que permiten tener una primera conclusión de la funcionabilidad y usabilidad del sistema. Estas preguntas se muestran a continuación.

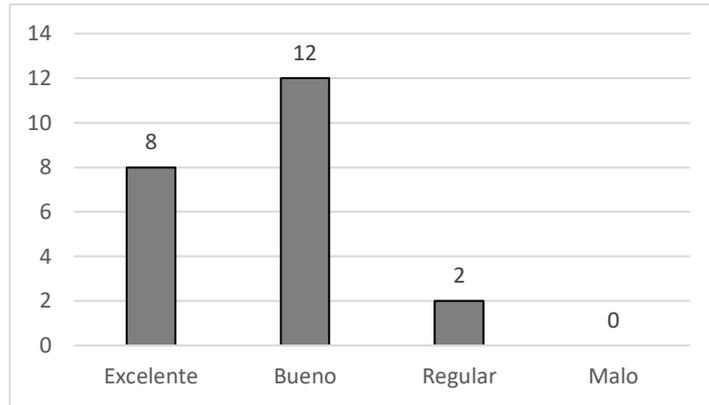
- Según su criterio: ¿qué tan amigable es la interfaz de control para el reconocimiento de gestos?



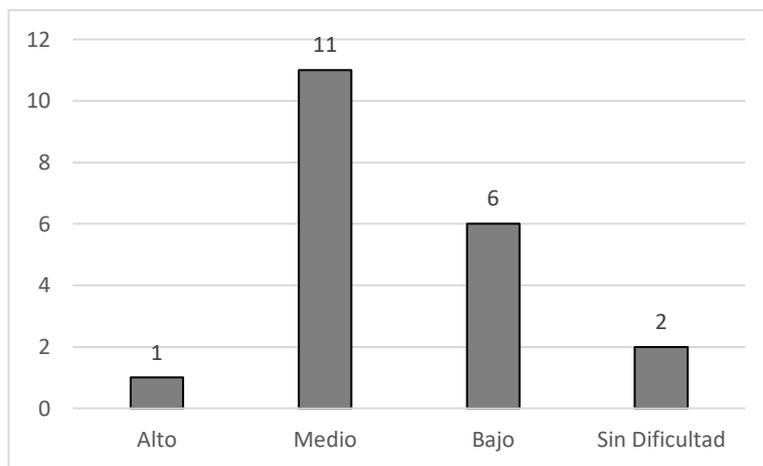
- Según su criterio valore la calidad (resolución, tamaño, nitidez) del sistema de realimentación visual desde la estación remota.



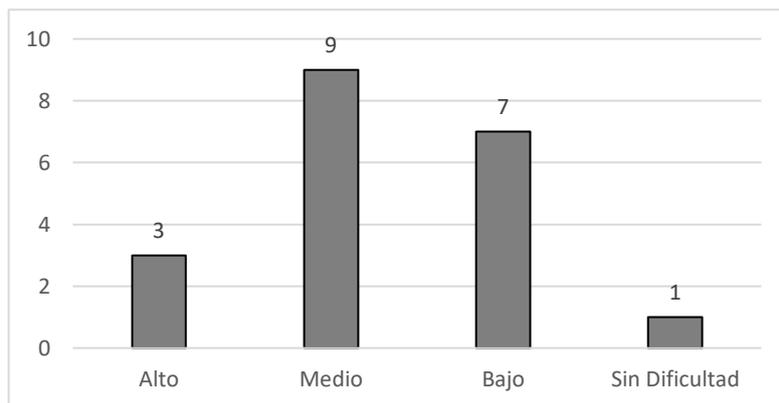
- Según su criterio valore qué tan cómodo se siente usando el brazalete (interfaz de control).



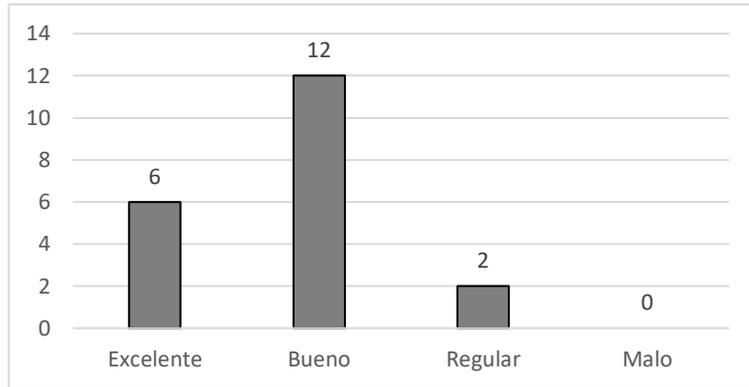
➤ ¿Qué tan difícil le resulta seleccionar la masa y color del objeto usando gestos?



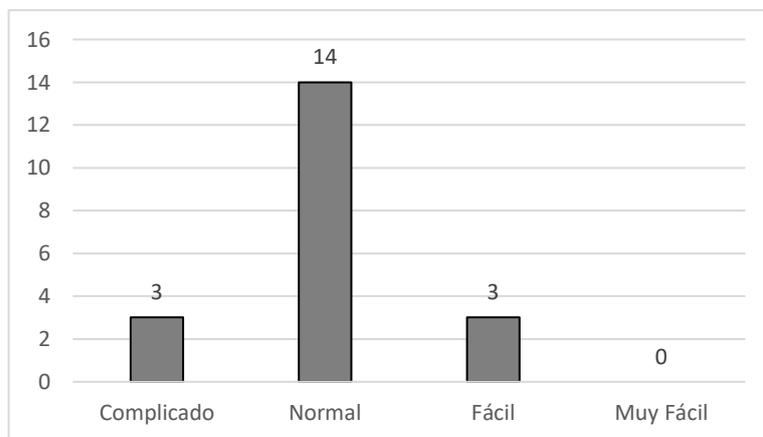
➤ ¿Qué tan difícil le resulta operar remotamente el brazo robótico virtual?



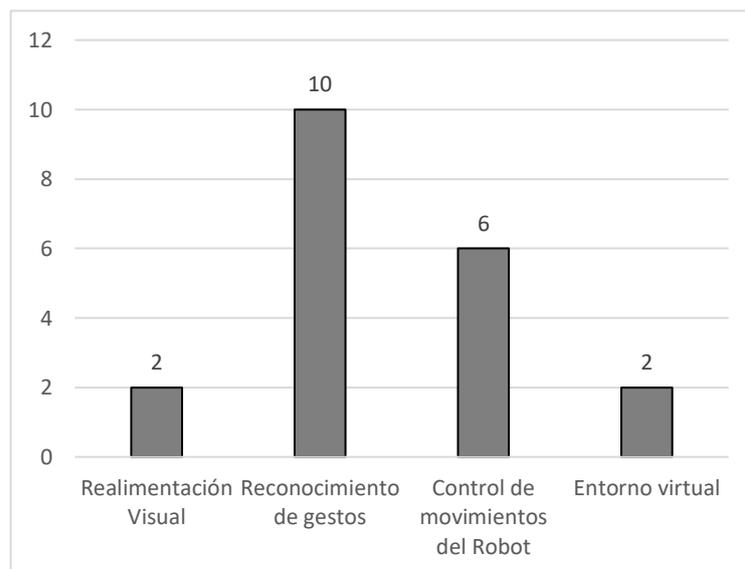
➤ ¿Qué tan rápido le parece el tiempo de respuesta desde que se le envía una acción, hasta que observa la misma?



➤ Según su criterio que tan maniobrable es el sistema en conjunto.



➤ Según su criterio ¿qué aspectos se puede mejorar en el sistema en conjunto?



- Finalmente existe una última pregunta se enfoca en recolectar comentarios del sistema, los cuales servirán de retroalimentación a los autores para un mejoramiento del sistema a futuro.

Dentro de las mejoras que se sugieren por parte de las personas involucradas en las pruebas, están las relacionadas con la mejora en identificación de gestos del sistema de reconocimiento de gestos durante el control del robot, ya que debido a su sensibilidad durante la operación del robot se reconocían inconscientemente los gestos. Así mismo, se sugiere aumentar la velocidad en los movimientos del robot simulado, siendo éste afectado por limitaciones de simulación.

Dentro de los comentarios se puede apreciar una buena acogida de los usuarios al sistema desarrollado en este proyecto, ya que, en algunos casos al darles explicaciones de las potenciales aplicaciones del sistema, éstos mostraron mucho interés en el mismo.

El resultado general del sistema, tomando en cuenta la encuesta realizada a los 20 usuarios, muestra que el sistema total en términos de usabilidad tiene un porcentaje de 82.91% de aceptación; mientras que en términos de funcionalidad el sistema total tiene un promedio de 64.79%. El bajo porcentaje de funcionalidad se debe a que los usuarios indican tener dificultad con la retroalimentación visual y necesitan más adaptación. De igual manera los usuarios manifiestan la necesidad de familiarizarse más tiempo con el reconocimiento de gestos antes de usarlo.

4. CONCLUSIONES Y RECOMENDACIONES

4.1. Conclusiones

- La deducción del modelo dinámico del manipulador ha permitido la implementación de controladores basados en las no linealidades del sistema, que para simplificar el sistema se genera un torque de control basado en las mismas no linealidades, como es en el caso de las fuerzas gravitacionales, las cuales son de gran influencia en manipuladores que se mueven en el plano vertical.
- El modelo dinámico ha permitido dentro de este proyecto la simulación de un robot virtual, siendo un aporte importante ya que desde el punto de vista práctico ayuda a probar esquemas de control sin la necesidad de tener el robot físico, o de ser el caso permite mejorar el desempeño de controladores antes de ser implementados en un robot real, evitando desgaste por estrés o posibles daños por errores del robot.
- El trabajo de modelamiento y simulación es una base que permitirá a futuro implementar un prototipo experimental, que ayude a validar y afinar el modelo dinámico obtenido para que éste, sea usado como un recurso didáctico en la enseñanza de temas relacionados con robótica, modelación y control.
- Desde el punto de vista teórico se ha verificado la potencialidad de controladores no lineales basados en el modelo como es el CPC, en el cual se evidencia una anulación “perfecta” de las no linealidades, siempre y cuando los parámetros del robot se conozcan de manera exacta, lo cual establece una limitación en su aplicación.
- La aplicación del controlador CIA, ha permitido generar un esquema que se acopla bastante bien para robots manipuladores, pues estos en la práctica al estar constantemente interactuando con herramientas u objetos, la dinámica del mismo se ve alterada, y como se ha comprobado por medio de los índices ISE e IAE para masas variables, el controlador se adapta bastante bien a las mismas anulando el error de seguimiento de trayectoria teniendo un mejor desempeño, mayor robustez y un rango de operación amplio.
- Todas las ecuaciones obtenidas del robot, pueden ser empleadas en trabajos a futuro, como por ejemplo un sistema de generación de trayectorias del robot en base a las ecuaciones de la cinemática inversa o también empleando las ecuaciones jacobianas.
- El clasificador SVM demora más de tiempo en dar una respuesta comparado con el clasificador árbol de decisiones o el del vecino más cercano, pero al buscar mayor

exactitud el clasificador SVM es el mejor clasificador por mostrar mejores resultados durante las pruebas.

- La efectividad total del sistema es del 96.1% usando 6 usuarios para la base de datos de entrenamiento y para la base de datos de testeo, y como se ha verificado el tiempo de respuesta del sistema general es menor a los 300 ms para ser considerado tiempo real, cumpliendo con uno de los objetivos del proyecto.
- El sistema de reconocimiento puede ser usado sin sesiones largas de entrenamiento, pues en el grupo de personas testeado se obtuvo un porcentaje de acierto del 91.6%, considerando que fueron personas ajenas al proceso de entrenamiento del sistema.
- El uso del brazalete MYO como elemento sensor facilita en gran medida la toma de señales y evita algunos artefactos producidos muchas veces por cables al ser un sistema inalámbrico. Además, se pudo verificar el uso de 2 brazaletes MYO para el control continuo del brazo robótico, siendo un aporte que puede ayudar en trabajos con otros dispositivos mecatrónicos con más grados de libertad.
- La prueba del sistema en conjunto permite apreciar la gran potencialidad que deja este trabajo, pues mediante el control remoto, el sistema de realimentación visual y una implementación del prototipo robótico a futuro, permitiría hacer uso del mismo en misiones de exploración que presenten un peligro para el operador.
- Durante las pruebas experimentales de operación remota, se puede concluir que la experiencia del usuario repercute en los resultados obtenidos, pues en un inicio existe dificultad en el control de movimiento del robot, misma que va disminuyendo conforme el usuario se acopla al sistema, logrando moverlo con mayor facilidad.

4.2. Recomendaciones

- Con relación al modelo del sistema robótico se recomienda que, en trabajos a futuros en conjunto con la implementación de un prototipo, se incluya la dinámica de los actuadores y de ser el caso las fricciones, pues son parámetros que se pueden determinar con el robot real.
- Además, dentro de los trabajos complementarios a futuro, se recomienda que se implemente los algoritmos de control sobre todo el CIA en sistemas embebidos para evaluar su desempeño en un prototipo, permitiendo de esta manera establecer mejoras y temas de investigación en el campo del control de manipuladores.
- Se recomienda trabajar en un sistema generador de trayectorias, con la finalidad que el sistema robótico sea usado en otros campos de aplicación e investigación como por ejemplo en el campo de la visión artificial.

- Se puede manejar un filtro digital con una frecuencia de corte mayor a 100 Hz para poder tener más información de las señales de EMG.
- Se puede tomar señales en diferentes posiciones del brazalete y tener un sistema más robusto e independiente de la colocación del brazalete.
- Se puede registrar más gestos de la mano, pero a la vez al sistema se vuelve más sensible por lo que se debe manejar la cantidad necesaria de gestos para no saturar al sistema con señales innecesarias que no se emplean en el control de algún dispositivo electrónico, mecánico.
- Se puede usar una retroalimentación auditiva en el caso que el ambiente del sistema no permita una retroalimentación visual, tubería, o tanques de almacenamiento.
- El sistema controlado puede ser empleado en diferentes áreas, medicina (rehabilitación), militar y de entretenimiento, sin embargo, existen parámetros que se deben tomar en cuenta como el tiempo de respuesta y el ambiente donde se trabaja pues en estos sistemas pueden ingresar artefactos que alteren la respuesta o comportamiento del sistema.

Referencias Bibliográficas

- [1] J. Carvajal, "Modelamiento y diseño de robots industriales", Universidad de la Salle.
- [2] A. Barrientos, L. Peñín, C. Balaguer, R. Aracil, "Fundamentos de robótica", 2da edición, McGraw-Hill, ISBN: 978-84-481-5636-7.
- [3] J. Craig, "Robótica", 3era edición, Pearson Prentice Hall, México 2006.
- [4] A. Ollero, "Robótica: Manipuladores y Robots Móviles", Marcombo S.A, España 2001.
- [5] K. Hernández, F. Panchi, "Diseño, Simulación e Implementación de un Sistema de Tele-operación para un Robot Móvil tipo Carlake", EPN-Ecuador, noviembre 2017.
- [6] R. Miranda, "Cinemática y Dinámica de Robots Manipuladores", 1era edición, Alfaomega, Mexico 2016, ISBN: 978-607-622-048-1.
- [7] F. Cortés, "Robótica Control de Robots Manipuladores", Primera edición, Alfaomega, México 2011, ISBN: 978-607-707-190-7
- [8] N. Munro, F. Lewis, "Robot Manipulator Control", 2da edición, Marcel Dekker, USA, 2004
- [9] V. Vimos, C. Yáñez, "Diseño e implementación de un sistema de control en tiempo real mediante ondas cerebrales", Escuela Politécnica Nacional, 2015.
- [10] A. Oña, "Implementación del Control de un Brazo Robótico Utilizando Sensores Incorporados en Brazales, para la Emulación del Movimiento de un Brazo Humano", Escuela Politécnica Nacional, 2014.
- [11] H. Maldonado, R. Silva, E. Ramos, V. Hernández, J. Rivera, "Modelado y simulación de un robot rígido de dos grados de libertad", Latin-American Journal of Physics Education, México 2011, ISSN.
- [12] N. Rosillo, A. Valera, F. Benimeli, V. Mata, "Real-Time Solving of Dynamics Problem in Industrial Robots", Industrial Robot: An International Journal, Volume 38, 2011.
- [13] S. Torres, J. Méndez, "Seguimiento de trayectorias en Robots Manipuladores: Revisión de Soluciones y Nuevas Propuestas", RIAI, 2009.
- [14] N. Kapoor, J. Ohri, "Sliding Mode Control (SMC) of Robot Manipulator via Intelligent Controllers", The Institute of Engineers, India 2016.
- [15] J. Cazalilla, M. Vallés, V. Mata, M. Díaz-Rodríguez, A. Valera, "Adaptive control of a 3-DOF parallel manipulator considering payload handling and relevant parameter models"
- [16] Urrea, E. C. y Kern, J., A New Model for Analog Servomotors. Practical Implementations, Canadian Journal on Automation, Control and Intelligent Systems 2, 29-38 (2011).
- [17] J. Medina, M. Nono, "Control de un Brazo Robótico para Clasificar Objetos Sólidos con Formas Definidas Utilizando Visión Estereoscópica", EPN-Ecuador, agosto 2012.

- [18] R. Gutiérrez, "Control de Manipuladores Teleoperados", Ciencia e Ingeniería Neogranadina, 2006.
- [19] J. Chico, "Desarrollo de Prototipo de Prótesis de Mano Emulada en un Ambiente Virtual", Universidad Nacional de Colombia, 2016
- [20] E. Nuño, "Teleoperación de Robots: Técnicas, Aplicaciones, Entorno Sensorial y Teleoperación Inteligente", España 2004. Disponible en: https://www.researchgate.net/publication/33421071_Teleoperacion_de_robots_tecnicas_aplicaciones_entorno_sensorial_y_teleoperacion_inteligente
- [21] P. K. Artemiadis and K. J. Kyriakopoulos, "Emg-based position and force control of a robot arm: Application to teleoperation and orthosis," in Advanced intelligent mechatronics, 2007 IEEE/ASME international conference on. IEEE, 2007, pp. 1{6.
- [22] D. Tan, T. S. Saponas, D. Morris, and J. Turner, "Wearable electroMYOgraphy based controllers for human-computer interface," May 1 2012, uS Patent 8,170,656.
- [23] S. Muceli and D. Farina, "Simultaneous and proportional estimation of hand kinematics from emg during mirrored movements at multiple degrees-of-freedom," IEEE transactions on neural systems and rehabilitation engineering, vol. 20, no. 3, pp. 371{378, 2012.
- [24] A. L. Edwards, M. R. Dawson, J. S. Hebert, C. Sherstan, R. S. Sutton, K. M. Chan, and P. M. Pilarski, "Application of real time machine learning to MYOelectric prosthesis control: A case series in adaptive switching," Prosthetics and orthotics international, vol. 40, no. 5, pp. 573{581, 2016.
- [25] F. V. Tenore, A. Ramos, A. Fahmy, S. Acharya, R. Etienne-Cummings, and N. V. Thakor, "Decoding of individuated movements using surface electroMYOgraphy," IEEE transactions on biomedical engineering, vol. 56, no. 5, pp. 1427{1434, 2009.
- [26] A. Nazemi and A. Maleki, "Artificial neural network classifier in comparison with lda and ls-svm classifiers to recognize 52 hand postures and movements," in Computer and Knowledge Engineering (ICCKE), 2014 4th International eConference on. IEEE, 2014, pp. 18{22.
- [27] M. E. Benalcázar, C. Motoche, J. A. Zea, A. G. Jaramillo, C. E. Anchundia, P. Zambrano, M. Segura, F. B. Palacios, and M. Pérez, "Real-time hand gesture recognition using the MYO armband and muscle activity detection," in Ecuador Technical Chapters Meeting (ETCM), 2017 IEEE. IEEE, 2017, pp. 16.
- [28] J. Kim, S. Mastnik, and E. Andre, "Emg based hand gesture recognition for real time biosignal interfacing," in Proceedings of the 13th international conference on Intelligent user interfaces. ACM, 2008, pp. 3039.
- [29] A. Fougner, S. Staudahl, P. J. Kyberd, Y. G. Losier, and P. A. Parker, "Control of upper limb prostheses: terminology and proportional MYOelectric control a review," IEEE Transactions on neural systems and rehabilitation engineering, vol. 20, no. 5, pp. 663{677, 2012.
- [30] E. Scheme and K. Englehart, "ElectroMYOgram pattern recognition for control of powered upper-limb prostheses: state of the art and challenges for clinical use." Journal of Rehabilitation Research & Development, vol. 48, no. 6, 2011.

- [31] M. T. Wolf, C. Assad, A. Stoica, K. You, H. Jethani, M. T. Vernacchia, J. Fromm, and Y. Iwashita, "Decoding static and dynamic arm and hand gestures from the jpl biosleeve," in Aerospace Conference, 2013 IEEE. IEEE, 2013, pp. 1{9.
- [32] C. Castellini, K. Hertkorn, M. Sagardia, D. S. Gonz_alez, and M. Nowak, "A virtual piano-playing environment for rehabilitation based upon ultrasound imaging," in Biomedical Robotics and Biomechatronics (2014 5th IEEE RAS & EMBS International Conference on. IEEE, 2014, pp. 548{554.
- [33] I. Kuzborskij, A. Gijsberts, and B. Caputo, "On the challenge of classifying 52 hand movements from surface electroMYOgraphy," in Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE. IEEE, 2012, pp. 49314937.
- [34] Weiss, L. D., Weiss, J. M., & Silver, J. K. (2015). Easy EMG: a guide to performing nerve conduction studies and electromyography. Elsevier Health Sciences.
- [35] Thalmic Labs Inc., Unveiling the Final Design of the MYO™ Armband, 10 junio 2014. [En línea]. Available: <http://blog.thalmic.com/unveiling-final-design-MYOarmband/>.
- [36] Duda, R., PE Hart y D. Stork, Patter Classification. New York: John Wiley. (2001).
- [37] Ken Sharman, A. E., "Interfaz Cerebro-Computadora", Grupo de Sistemas Adaptativos Complejos @ ITI, Enero 2004
- [38] Farfán F. D., Politti J. C., Felice C. J., "Evaluación de Patrones Temporales y Espectrales para el Control Bioeléctrico", Departamento de Bioingeniería – FACET – UNT. ffarfan@herrera.unt.edu.ar.
- [39] Guerra Hernández A., Aprendizaje Automático: Clasificación, Universidad Veracruzana, Facultad de Física e Inteligencia Artificial, Maestría en Inteligencia Artificial, Enero 2004.
- [40] Lorca P., j. de andrés, j. díez, j. j. del coz, a. bahamonde. el análisis de preferencias: un nuevo enfoque para el estudio de la rentabilidad empresarial, Universidad de Oviedo, España; vol. XXXI (2), 2007.
- [41] R. Liye, W. Xiaoli and W. Xiao. "Research of Feature Extraction Method for Stroke Patients Surface ElectroMYOgraphy". Intelligent Networks and Intelligent Systems (ICINIS). pp. 322-324. 2012
- [42] G. Yang and S. Yang. "Emotion Recognition of ElectroMYOgraphy Based on Support Vector Machine". Intelligent Information Technology and Security Informatics (IITSI). pp. 298-301. 2010
- [43] A. Salman, J. Iqbal, U. Izhar, U.S. Khan and N. Rashid, "Optimized circuit for EMG signal processing". Robotics and Artificial Intelligence (ICRAI), 2012 International Conference. pp. 208-213. 2012
- [44] K. HeeSu, R. Kiwon, Y. Kyoung-jin, S. Hyun-chool, "Intuitive robot navigation using wireless EMG and acceleration sensors on human arm". Intelligent Signal Processing and Communications Systems (ISPACS), 2011 International Symposium, vol. 7(9), pp. 1-4 Dec. 2011.

- [45] W. Xinqing, L. Yiwei, Y. Dapeng, L. Nan, L. Jiang, L. Hong, "Progress in the biomechatronic design and control of a hand prosthesis". Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference, pp. 5880-5885. 2010
- [46] Estudio de la robótica y desarrollo tecnológico actual en Ecuador (2018). Obtenido de <https://www.eluniverso.com/noticias/2018/08/17/nota/6909632/era-robot-se-instala-paso-lento-ecuador>
- [47] Implementación de algoritmos para la extracción de patrones característicos en Sistemas de Reconocimiento De Voz en Matlab - David Reig Albiñana-Politecnica de Valencia 2014
- [48] Farina, D., Jiang, N., Rehbaum, H., Holobar, A., Graitmann, B., Dietl, H., & Aszmann, O. C. (2014). The extraction of neural information from the surface EMG for the control of upper-limb prostheses: emerging avenues and challenges. IEEE Transactions on Neural Systems and Rehabilitation Engineering, 22(4), 797-809.
- [49] Descripción de los músculos del antebrazo y sus principales funciones. Obtenido de <http://max-flores-gamboa.blogspot.com/2012/04/musculos-del-miembro-superior.html>
- [50] Luc Devroye, László Györfi, Gábor Lugosi, "A Probabilistic Theory of Pattern Recognition", Springer 1996.

Anexos

ANEXO I

ENCUESTA REALIZADA A 20 USUARIOS PARA EVALUAR LA FUNCIONALIDAD Y USABILIDAD DEL SISTEMA

TEST DE RECONOCIMIENTO DE GESTOS

Nombre: _____
Edad: _____
Estatura aprox: _____
Peso aprox: _____

Dario Sunkari

1 Según su criterio en el test, ¿qué tan bien el sistema reconoce el gesto Wave-out?

Marque con una X

- 1 Excelente
- 2 Bueno
- 3 Regular
- 4 Malo



2 Según su criterio en el test, ¿qué tan bien el sistema reconoce el gesto Wave-in?

Marque con una X

- 1 Excelente
- 2 Bueno
- 3 Regular
- 4 Malo



3 Según su criterio en el test, ¿qué tan bien el sistema reconoce el gesto Close?

Marque con una X

- 1 Excelente
- 2 Bueno
- 3 Regular
- 4 Malo



4 Según su criterio en el test, ¿qué tan bien el sistema reconoce el gesto Thumb?

Marque con una X

- 1 Excelente
- 2 Bueno
- 3 Regular
- 4 Malo



5 Según su criterio en el test, ¿qué tan bien el sistema reconoce el gesto Relax?

Marque con una X

- 1 Excelente
- 2 Bueno
- 3 Regular
- 4 Malo



6 Según su criterio en el test, ¿qué tan bien el sistema reconoce el gesto Index?

Marque con una X

- 1 Excelente
- 2 Bueno
- 3 Regular
- 4 Malo



- 7 Según su criterio: ¿qué tan amigable es la interfaz de control para el reconocimiento de gestos?
- Marque con una X
- | | |
|-------------|-------------------------------------|
| 1 Excelente | <input type="checkbox"/> |
| 2 Bueno | <input checked="" type="checkbox"/> |
| 3 Regular | <input type="checkbox"/> |
| 4 Malo | <input type="checkbox"/> |
- 8 Según su criterio valore la calidad (resolución, tamaño, nitidez) del sistema de realimentación visual desde la estación remota.
- Marque con una X
- | | |
|-------------|-------------------------------------|
| 1 Excelente | <input type="checkbox"/> |
| 2 Bueno | <input checked="" type="checkbox"/> |
| 3 Regular | <input type="checkbox"/> |
| 4 Malo | <input type="checkbox"/> |
- 9 Según su criterio valore qué tan cómodo se siente usando el brazalete (interfaz de control)
- Marque con una X
- | | |
|-------------|-------------------------------------|
| 1 Excelente | <input type="checkbox"/> |
| 2 Bueno | <input type="checkbox"/> |
| 3 Regular | <input checked="" type="checkbox"/> |
| 4 Malo | <input type="checkbox"/> |
- 10 ¿Qué tan difícil le resulta seleccionar la masa y color del objeto usando gestos?
- Marque con una X
- | | |
|------------------|-------------------------------------|
| 1 Alto | <input type="checkbox"/> |
| 2 Medio | <input checked="" type="checkbox"/> |
| 3 Bajo | <input type="checkbox"/> |
| 4 Sin dificultad | <input type="checkbox"/> |
- 11 ¿Qué tan difícil le resulta operar remotamente el brazo robótico virtual?
- Marque con una X
- | | |
|------------------|-------------------------------------|
| 1 Alto | <input type="checkbox"/> |
| 2 Medio | <input checked="" type="checkbox"/> |
| 3 Bajo | <input type="checkbox"/> |
| 4 Sin dificultad | <input type="checkbox"/> |
- 12 ¿Qué tan rápido le parece el tiempo de respuesta desde que se le envía una acción, hasta que observa la misma?
- Marque con una X
- | | |
|-------------|-------------------------------------|
| 1 Excelente | <input type="checkbox"/> |
| 2 Bueno | <input type="checkbox"/> |
| 3 Regular | <input checked="" type="checkbox"/> |
| 4 Malo | <input type="checkbox"/> |
- 13 Según su criterio evalúe la complejidad de mover el objeto seleccionado de un punto a otro.
- Marque con una X
- | | |
|------------------|-------------------------------------|
| 1 Alto | <input type="checkbox"/> |
| 2 Medio | <input checked="" type="checkbox"/> |
| 3 Bajo | <input type="checkbox"/> |
| 4 Sin dificultad | <input type="checkbox"/> |
- 14 Según su criterio evalúe ¿qué tan fácil le resulta interactuar con el brazo robótico en el entorno virtual?
- Marque con una X
- | | |
|-------------|-------------------------------------|
| 1 Muy bueno | <input type="checkbox"/> |
| 2 Bueno | <input checked="" type="checkbox"/> |
| 3 Regular | <input type="checkbox"/> |
| 4 Malo | <input type="checkbox"/> |
- 15 Según su criterio que tan maniobrable es el sistema en conjunto.
- Marque con una X
- | | |
|--------------|-------------------------------------|
| 1 Complicado | <input type="checkbox"/> |
| 2 Normal | <input checked="" type="checkbox"/> |
| 3 Fácil | <input type="checkbox"/> |
| 4 Muy Fácil | <input type="checkbox"/> |

16 Según su criterio ¿qué aspectos se puede mejorar en el sistema en conjunto?

Marque con una X

- 1 Realimentación Visual
- 2 Reconocimiento de gestos
- 3 Control de movimientos del Robot
- 4 Entorno virtual

<input type="checkbox"/>
<input checked="" type="checkbox"/>
<input type="checkbox"/>
<input type="checkbox"/>

17 Comentario o sugerencia adicional respecto a su experiencia utilizando el sistema (opcional)

En el entorno virtual, debe reconocer de manera eficiente los movimientos realizados.

Intento	Out	In	Close	Index	Thumb
1	✓	✓	✓	✓	✓
2	✓	✓	✓	—	✓
3	✓	—	✓	✓	✓
4	✓	✓	✓	✓	✓
5	✓	—	✓	✓	✓
6	✓	✓	✓	—	✓
7	✓	✓	✓	✓	✓
8	✓	✓	✓	✓	✓
9	✓	—	✓	✓	✓
10	✓	✓	✓	✓	✓

$t_1 = 23 \text{ seg.}$

$t_2 = 23 \text{ seg.}$