

## Sistema de reconocimiento de gestos para un robot de cuatro grados de libertad

Fernández Ramírez Arnoldo, Olivares Borja Eduardo, Macías Solís Francisco, García Andrade Roxana<sup>1,a</sup>, Martínez Reyes Armando<sup>1,b</sup>, Rincón Martínez Ernesto.

Tecnológico Nacional de México / Instituto Tecnológico de Nuevo León  
División de Estudios de Posgrado e Investigación  
Departamento de Ingeniería Eléctrica y Electrónica 1  
Av. Eloy Cavazos # 2001 Col. Tolteca C.P. 67170 Guadalupe Nuevo León México  
Tel.+ 81 8157 0500 ext 159  
olivareseduardo97@gmail.com

### Resumen

En el presente trabajo se realizó un estudio acerca de las interacciones que existen entre hombre – Robot, por medio del reconocimiento de gestos, se desarrolló el diseño de un sistema que permite gestionar un robot de cuatro grados de libertad de manera remota con movimientos del cuerpo humano, mediante el uso del sensor *leap motion*, es posible que el usuario, interactúe de manera natural, solo moviendo alguna extremidad del cuerpo humano (mano y antebrazo) sin descuidar otros aspectos importantes en la conducción del robot. La parte remota de la interacción es basada en módulos de comunicación *Xbee*, estos son radios que permiten la comunicación inalámbrica entre dispositivos electrónicos, en este caso *Arduino*, el cual permite la manipulación a distancia de nuestro sistema. Finalmente, con el propósito de concluir el proyecto se buscaron las áreas de oportunidad en las diversas líneas de investigaciones de proyectos similares ya existentes.

**Palabras clave**— control gestual, interacción, teleoperación, robot.

### Abstract

*The present project carries out a study about the interactions that exist between man - Robot, through the use of gesture recognition, the design of a system that allows to manage a robot of four degrees of freedom remotely with movements of the human body is explained, by using the Leap Motion sensor, it is possible for the user to interact naturally, only by moving some limb of the human body (hand and forearm) without specifying other important aspects in the robot's driving. The remote part of the interaction is based on Xbee communication modules that are radios that allow wireless communication between electronic devices, in this case Arduino, which allows the distance of our system distance. Finally, with the purpose of concluding the search project in the areas of opportunity in the various research lines of similar and dangerous projects.*

**Keywords**— gesture control, interaction, teleoperation, robot.

## 1. INTRODUCCIÓN

La primera interacción considerable con un dispositivo de control de una máquina, que experimento el hombre fue el interruptor y los pulsadores, como por ejemplo para hacer determina función, apagar/encender la máquina, activar una luz, o un movimiento. La tecnología siguió evolucionando y esas mismas funciones fueron reemplazadas por la tecnología táctil, en la cual, tocando una superficie se realizaban las mismas funciones anteriormente descritas.

Pero las dos tecnologías presentadas anteriormente tienen diversos inconvenientes, el principal es que tienes que tocar el interruptor/pulsador o la pantalla táctil para poder interactuar con la máquina, con el riesgo que puede tener el usuario, como, por ejemplo, el tocarla, puesto que puede estar cargada eléctricamente o existir la posibilidad de no poder interactuar con la máquina (robot), entre otras contingencias. Por lo cual el reconocimiento gestos en los últimos años ha sido objeto de múltiples investigaciones en la robótica por la capacidad de explotar todo el rango de movimiento en 3D para controlar robots de manera simple basada en movimientos de la mano (traslación y rotación).

Actualmente con los avances en ciertos campos de la electrónica, es de sorprender contar con la opción de controlar sistemas ciberfísicos utilizando el control gestual y poder interactuar con toda la arquitectura de sistema, con los diversos dispositivos que se encuentran en el mercado.

Estas tecnologías han encontrado diversas aplicaciones en el campo de la ingeniería algunos ejemplos son en aplicaciones médicas, como la generación programas de entrenamiento para personas con discapacidad, exoesqueletos y dispositivos robóticos para rehabilitación física, aplicaciones de aprendizaje como sistemas interactivos, cuyo objetivo es hallar una interacción entre humanos y tecnología más intuitiva.

Pero la efectividad es importante para el funcionamiento de este sistema, por lo cual se escogió el dispositivo con el tiempo de respuesta más preciso, aumentando la interacción y teniendo la posibilidad de reconocer todos los gestos sin fatigar al usuario.

El objetivo de este trabajo es desarrollar un sistema de reconocimiento de gestos para un robot de cuatro grados de libertad, que mediante los gestos corporales (mano y antebrazo) permita representar un movimiento, ver fig.1.

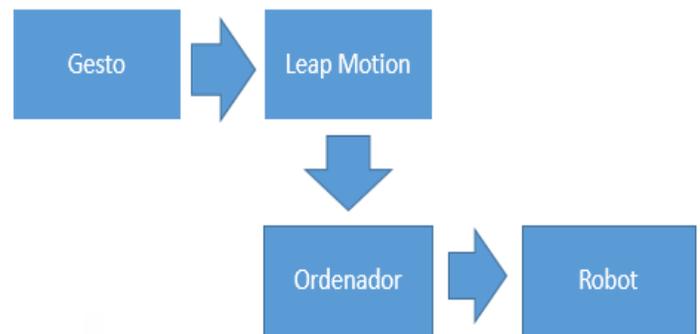


fig.1 Diagrama de bloques del sistema.

Este trabajo está organizado de la siguiente manera, en la sección 2 se resume todos los trabajos de investigaciones similares, posteriormente en la sección 3 se observará la metodología utilizada para el diseño, el desarrollo y la integración del sistema. En la sección 4, Se finaliza con las conclusiones de la aplicación desarrollada y las referencias consultadas.

## 2. ESTADO DEL ARTE

En esta sección se describen los detalles de investigaciones relacionadas con los temas de reconocimiento de gestos, robótica, control gestual, interacción hombre-máquina y teleoperación, con la finalidad de encontrar el mejor método para el desarrollo del sistema. De esta forma, en [1] indica en su trabajo la eficiencia de algunos dispositivos de movimiento, dentro de los cuales, se encuentra el brazalete mioeléctrico (*Myo*), *Kinect* y *Leap Motion*, donde se comprobó cuál de los 3 sensores ya antes mencionado tenía mayor eficiencia en el reconocimiento de gestos obteniendo el mejor resultado el dispositivo *Leap Motion* con el 90% de efectividad. Se obtuvo la información de la eficiencia en el siguiente estudio [2] este detalla el desarrollo de un sistema de rehabilitación basado en *Kinect*, que puede guiar de manera más efectiva e intuitiva, pero a la vez explica la deficiencia que tiene el sensor con los movimientos de las articulaciones en el usuario. Por otra parte, en [3] propone un sistema basado en ROS para reconstruir el movimiento del miembro superior humano basado en datos recogidos con el brazalete *Myo* observando que tiene un margen de error en la mayoría de sus gestos propuestos. De la misma manera, en [4] presenta el diseño y construcción de un Guante de Interacción Natural de Usuario para el reconocimiento de gesto incorporando distintos tipos de sensores (IMU y sensores resistivos). El inconveniente son los falsos contactos eléctricos en el guante, el cual provoca fallas en el dispositivo impidiendo reconocer el gesto y el correcto funcionamiento del guante. Del mismo modo, en [5] redacta un sistema de control gestual de un brazo robótico (SCARA) que incluye los gestos mover, agarrar y girar con un alto desempeño en la captura de los gestos. De igual manera, en [6] describe una evaluación de aplicaciones interactivas de gestos, usando el sensor *Leap Motion* para la visualización y pruebas de usabilidad, verificando los requisitos importantes como la precisión de los gestos, el tiempo de interacción, el nivel de interactividad, la naturalidad, la eficacia, la facilidad de uso, la calidad visual de interfaz, la utilidad, la satisfacción y la no ocurrencia de fatiga indicando su fuerte potencial como herramienta de apoyo y libre contacto. También se investigó diferentes tipos de protocolo de comunicación inalámbrica como *ZigBee*, como lo menciona el siguiente artículo [7], donde se explica el diseño y fabricación de un sistema de seguridad para una monta cargas utilizando módulos de comunicación inalámbricos *Xbee* para notificar a los operadores la ubicación general de peatones cercanos. Los resultados de las pruebas muestran que el sistema puede funcionar en cualquier otro proyecto que necesite mando a distancia o aplicaciones similares.

## 3. METODOLOGIA

### 3.1 Diseño

A continuación, se muestra en la fig.2, un esquema físico de la solución inicial del proyecto que nos permitirá ver las partes que conforman la arquitectura del sistema.

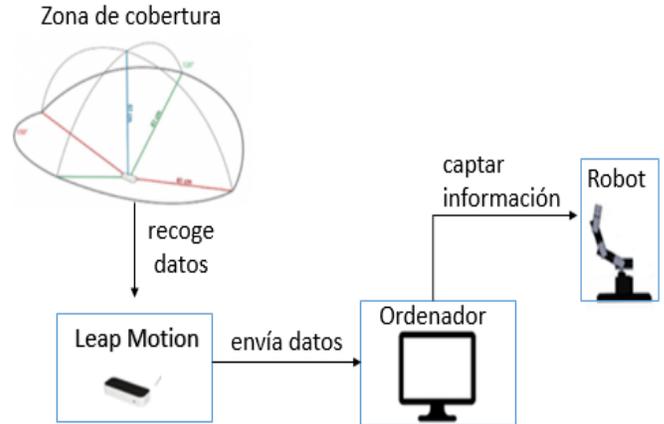


fig.2 Esquema de solución inicial.

Está conformado principalmente por el sensor *Leap Motion* encargado de recoger los gestos del usuario y enviar los datos recaudados al ordenador, el cual realiza la tarea de procesar las imágenes para que después nuestra tarjeta de adquisición de datos (*Arduino*) obtenga la posición de las manos, recibida de forma inalámbrica para posteriormente mover el robot.

El sensor utilizado para la captura de gestos, es el *Leap Motion* por sus especificaciones técnicas ya que, es un dispositivo con dimensiones muy reducidas (75 mm de largo, 25 mm de ancho y 11 mm de alto.) y cuenta con dos cámaras con sensor monocromático, sensible a la luz infrarroja que puede trabajar a una velocidad de hasta 200 fotogramas por segundo. además de contar con un circuito integrado encargado de recoger la información de los sensores para luego enviarla al driver o controlador instalado en el ordenador [8]. ver fig.3

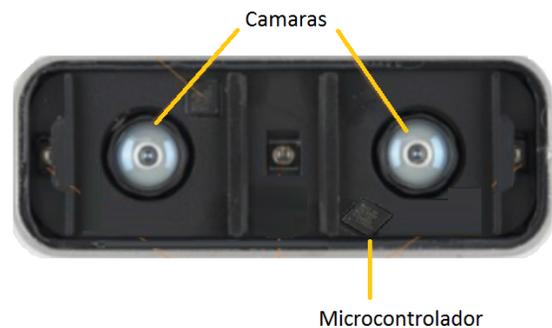


Fig.3 Dispositivo *Leap Motion*.

La adquisición de datos del sistema, se realizó utilizando la tarjeta *Arduino*, visto que, es de bajo costo, fácil de utilizar y de Alta flexibilidad. Además, cuenta con las principales especificaciones para procedimiento propuesto, como lo son, el voltaje operativo a 5v, pines digitales de entrada y salida (de las cuales 6 son de salida PWM) y es versátil con todas las plataformas informáticas[9]. Ver fig4

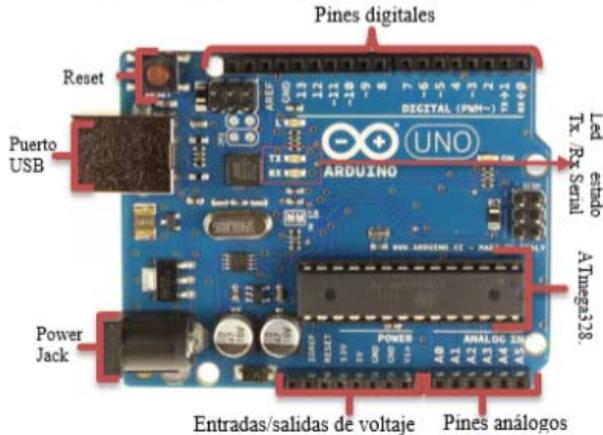


Fig.4 Ilustración representativa del *Arduino*.

Para la conexión inalámbrica se utilizó módulos *Xbee* que permite la creación de redes a 40 metros de cobertura en zona urbana, son de bajo consumo de energía y además poseen pines digitales y analógicos para la conexión de sensores y actuadores. En la fig.4 se muestra una ilustración del módulo *Xbee*[10].



Fig.5 Ilustración del módulo *Xbee*.

El robot utilizado en esta investigación es un mini-robot antropomórfico de cuatro grados de libertad realizado en el Instituto Tecnológico de Nuevo León y cuenta con 4 servomotores de 9 gramos de torque con dimensiones pequeñas (22 x 12.1 x 28.8 mm) y un diseño semejante a la de un dedo humano con el propósito de diseñar en trabajos futuros una extremidad fundamental del cuerpo humano (mano). A continuación, en la Fig.6 se muestra el primer prototipo del robot de 4 grados de libertad.



Fig.6 Robot de cuatro grados de libertad.

### 3.2 Desarrollo e Integración del Sistema

Se desarrolló un algoritmo que permitiera generar movimientos en el robot a través de señales de la mano. El sistema fue desarrollado en las plataformas de *Windows*, *Arduino* y *Processing*. Ver fig.7

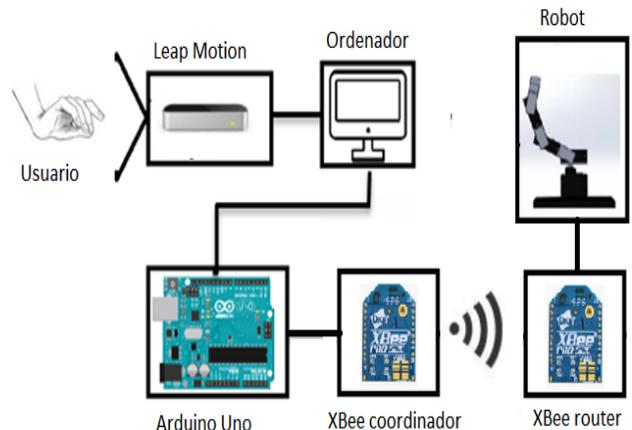


Fig.7 Esquema del sistema físico.

El movimiento del robot es en tiempo real y se ejecuta en el sketch de *Processing*, cuando el sensor de movimiento detecta la mano abierta. De lo contrario el código en *Arduino*, no se ejecuta y el robot no se mueve. en la fig.8 se muestra un diagrama de flujo del funcionamiento del procedimiento.

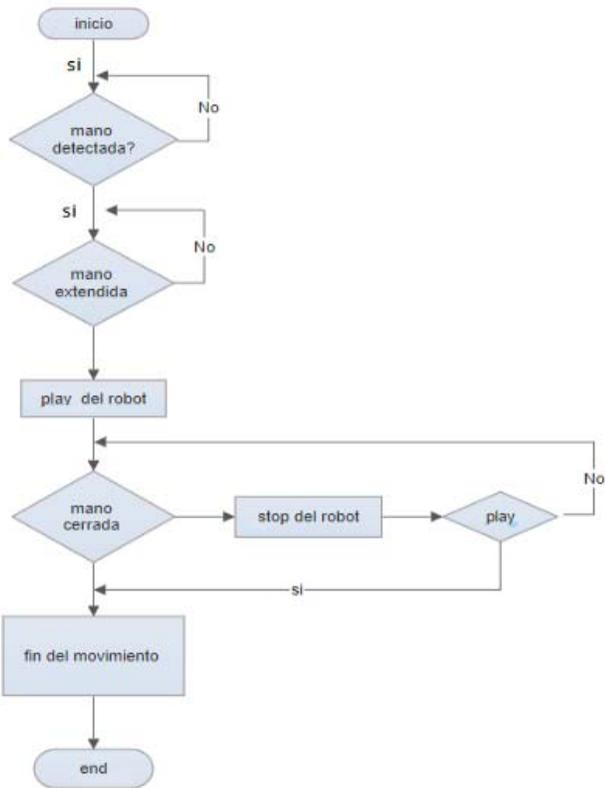


Fig.8. Diagrama de flujo del programa.

El uso de la tarjeta *Arduino* decide el número de grados que debe girar cada actuador, para que el robot alcance la posición deseada, esta condición es dada por el desarrollador del código. A continuación, se muestra en la fig.9, un esquema general del comportamiento de envío de datos desde la interfaz creada en *Processing* a la plataforma de *Arduino* y *Xbee*.

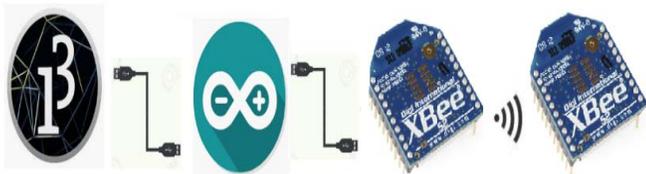


Fig.9 Esquema general del comportamiento de envío de datos.

En la siguiente fig.10, se muestra un esquema más detallado para establecer la conexión serial del módulo *Xbee* (Coordinador) con el microcontrolador *Arduino*, que será el vínculo entre el ordenador y los módulos *Xbee*. La conexión es a través de los pines TX y RX en ambos módulos [10].

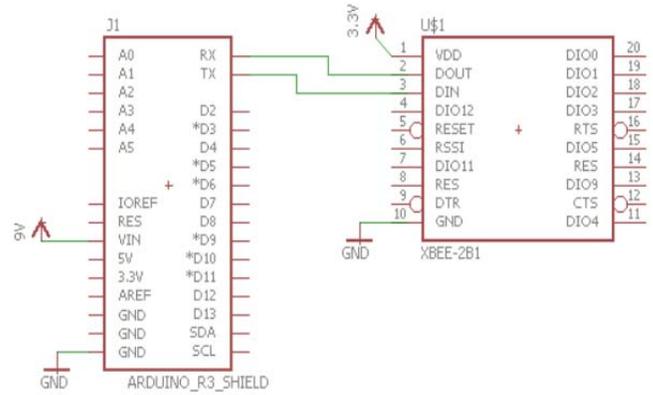


Fig.10 Esquema de conexión *Arduino* y *Xbee* (Coordinador).

El tipo de topología utilizada en nuestra método, fue el de punto ya que solo utilizaremos un coordinador y un router, para la comunicación inalámbrica [11] ver fig.11

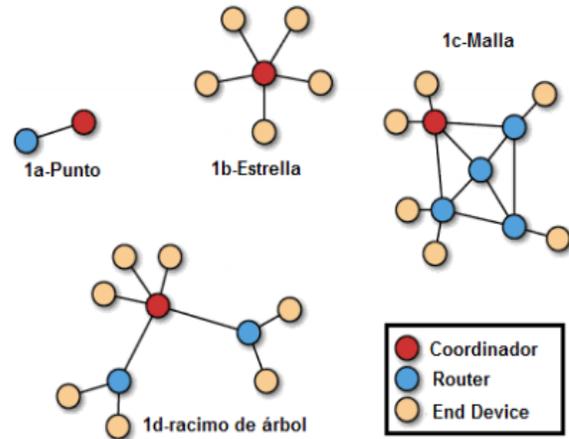


Fig.11 Topología *ZigBee*.

En la siguiente Tabla 1.se muestra la parte del código en *Arduino*. donde se accede a los métodos de la librería *VarSpeedServo.h* para controlar la velocidad y el giro de los servomotores y hacemos uso de los métodos *attach*, que nos permite indicar en que pin digital tenemos conectado nuestros servomotores, por otro parte el ángulo queremos posicionar nuestros servomotores es a través del método *write*. Para posteriormente en el *void loop* declarar las estructuras de control *if* que verifica una condición y la ejecuta previamente con la función *map* (), el cual no hace un filtrado de las posiciones en grados que permite re-mapear valores de un rango de entrada, a valores de un rango de salida[12].

Tabla1 Código de Arduino.

```
#include <VarSpeedServo.h>
VarSpeedServo falange1;
VarSpeedServo falange2;
VarSpeedServo falange3;

int num;
int mov;

// constructores
void setup () {
  Serial.begin(9600);
  falange1.attach(9);
  falange1.write(85,150) ;// motor de abajo

  falange2.attach(10) ;//motor del medio
  falange2.write(150,150);

  falange3.attach(13) ;// motor de arriba
  falange3.write(90,150);
;
}

//funciones miembro
void loop () {
  if (Serial.available()>0) {
    num = Serial.read();
    if (num >=0 && num <110) {
      mov = map (num,0, 109,90, 110);
      falange1.write(mov);
    }

    if (num >=0 && num <150) {
      mov = map (num, 0, 149, 90, 150);
      falange2.write(mov);
    }

    if (num >=0 && num <166) {
      mov= map (num, 0, 165, 80, 166);
      falange3.write(mov);
    }
  }
}
```

La interfaz consta de una ventana donde se muestra la simulación de un robot de cuatro grados de libertad formada por polígonos regulares, el cual se puede ver en la fig.12 la interfaz cuenta con siete vistas diferentes (frontales, lateral derecha, lateral izquierda, superior, inferior, posterior e isométrica), además en la parte central izquierda de la pantalla se despliega una ventana de la cámara web donde se puede observar el usuario o bien colocar el robot. Del mismo modo se puede acceder a una página de internet que muestra información del sensor por medio de un botón que se encuentra en la parte superior izquierda de la ventana[13]. Ver fig.13.

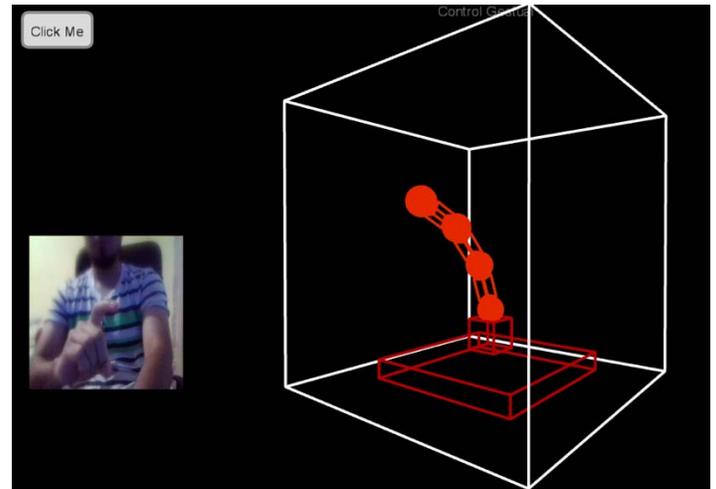


Fig.12 Simulación del robot.

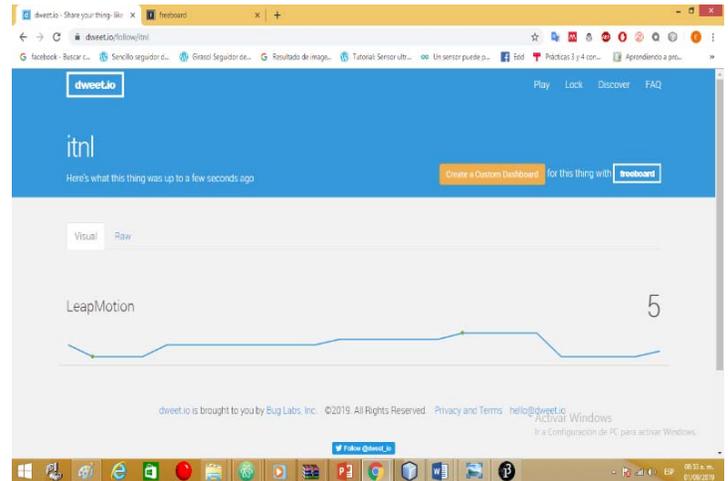


Fig.13 Ilustración de la página.

El tipo de gesto implementado en la interfaz tiene el nombre de “Swipe”, este es un movimiento en línea del mano extendido. Este gesto se detecta por dedo, por tal motivo, esto permite detectar si la mano completa se desliza de derecha a izquierda o viceversa, de igual manera de arriba hacia abajo o viceversa. ver fig.14.

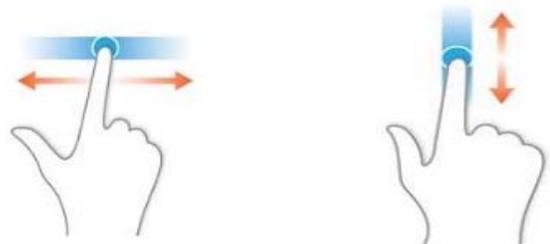


Fig.14 Ilustración del gesto implementado.

Los gestos anteriores solo hacen mención a dos dimensiones, en el plano “X” (derecha - izquierda) y el plano “Y” (arriba - abajo), mediante la siguiente función llamada “*leap*” nombrada por el desarrollador del código. En este caso, para diferenciar el tipo de deslizamiento se accedió al método “posición ()” del SDK de la aplicación de *leap motion* para comparar los valores absolutos de “X” o “Y”, de modo que, entre estos mismos se pueda saber si el desplazamiento fue horizontal o vertical, e interpretarse dentro de la interfaz [8].

#### 4. Validación y pruebas

El operador se posiciona frente el sensor de movimiento, realiza el primer gesto del robot que es flexión y extensión (arriba - abajo) como se puede observar en la fig.15, también se mostró en la fig.16 la respuesta de cada servomotor, cuando se activa. Se acciona mediante los 6 gestos que reconoce: el algoritmo (arriba, abajo, izquierda, derecha, puño cerrado y palma abierta). Cuando la palma está abierta empieza a mandar señales adquiridas de estos gestos y se obtienen la posición que realizan los 4 servomotores para que el robot genere los movimientos específicos ya antes mencionados en el código de la Tabla 1.

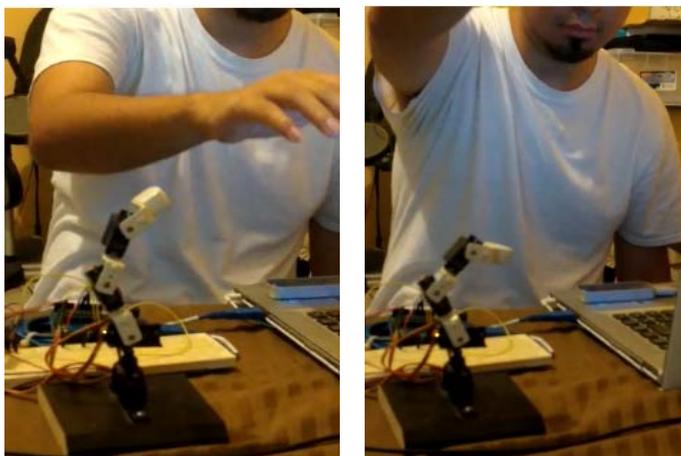


Fig.15.- A) extensión

B) flexión

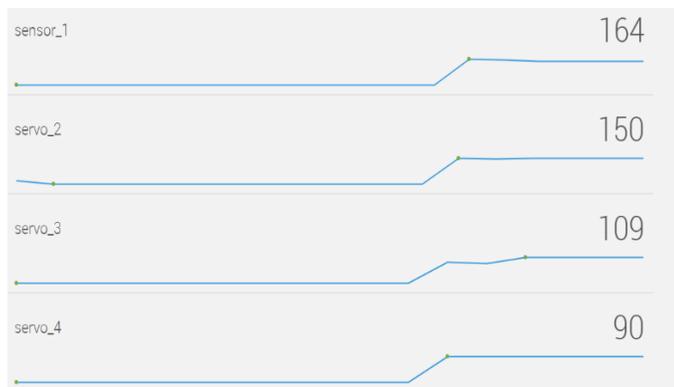


Fig.16 Grafica de los servomotores

#### 5. Conclusión.

En conclusión, se puede lograr la realización de aplicaciones intuitivas, que puedan mejorar la comunicación entre el

usuario y el robot, logrado generar una interacción de teleoperación entre emisor y receptor de manera más natural. Los resultados experimentales muestran que el sistema experimental logró reconocer los gestos con precisión. Además de presentar una solución satisfactoria para mando a distancia.

El trabajo futuro abordará la calibración conjunta de dos dispositivos de movimiento con el fin de calcular nuevas características basadas en la combinación de las posiciones 3D y una mejora al diseño del robot de cuatro grados de libertad.

#### 6. Referencias.

- [1] L. Vokorokos, J. Mihal’Ov, and E. Chovancova, “Motion sensors: Gesticulation efficiency across multiple platforms,” INES 2016 - 20th Jubil. IEEE Int. Conf. Intell. Eng. Syst. Proc., pp. 293–298, 2016.
- [2] A. Chiang, Q. I. Chen, and Y. A. O. Wang, “Kinect-Based In-Home Exercise System for Lymphatic Health and Lymphedema Intervention,” IEEE J. Transl. Eng. Heal. Med., vol. 6, no. November, pp. 1–13, 2018.
- [3] S. Tortora, “Dual-Myo Real-Time Control of a Humanoid Arm for Teleoperation,” 2019 14th ACM/IEEE Int. Conf. Human-Robot Interact., pp. 624–625, 2019.
- [4] J. A. Jiménez-montañés and G. P. J. A. Holgado-terriz, “El gesto aplicado al control de dispositivos en tiempo real usando un guante de interacción de bajo costo,” no. October, 2017.
- [5] C. Chen, L. Chen, X. Zhou, and W. Yan, “Controlling a Robot Using Leap Motion,” 2017.
- [6] E. S. S. Maria Andrea Formico, “Gesture Interaction and Evaluation Using the Leap Motion for Medical Visualization,” 2015.
- [7] J. Moseley, J. Howell, and S. Lim, “A Novel Forklift Safety System Utilizing Xbees and Directional Microstrip Patch Antennas,” pp. 3–4, 2017.
- [8] “Leap Motion.” [Online]. Available: <https://www.leapmotion.com/en/>. [Accessed: 13-Jun-2018].
- [9] J. Misael and S. Tarija, “Características de las placas arduino,” vol. 2, pp. 2–7, 2017.
- [10] P. Aguayo, “Guía del Usuario XBee Series 1,” pp. 1–69, 2008.
- [11] C. Alberto, V. Romero, J. Erickson, B. Jaimes, D. Carolina, and P. González, “Tecnura,” vol. 19, no. 45, pp. 141–157, 2015.
- [12] “Arduino - BuiltInExamples.” [Online]. Available: <https://www.arduino.cc/en/Tutorial/BuiltInExamples>. [Accessed: 01-Sep-2019].
- [13] “Examples \ Processing.org.” [Online]. Available: <https://processing.org/examples/>. [Accessed: 01-Sep-2019].