

Desarrollo de un Brazo Robótico e implementación de una estrategia de control por reconocimiento de gestos mediante Leap Motion

Jesús Sergio Artal y José Luis Montañés

Departamento Ingeniería Eléctrica. Escuela de Ingeniería y Arquitectura EINA.
Universidad de Zaragoza, Campus Río Ebro. María de Luna nº 3.
Edificio Torres Quevedo, 50018. Zaragoza, Spain.
E-mail: jsartal@unizar.es. Tfno: 976 762823.

Abstract— El objetivo del presente documento ha sido el desarrollo de un prototipo de brazo robótico articulado y la implementación de un método de control mediante el reconocimiento de gestos (Leap Motion Sensor) a través del movimiento natural del antebrazo y mano. La serie de avances relativos a las técnicas de control han provocado que la robótica se haya introducido también como complemento educativo en las enseñanzas básicas obligatorias. El trabajo Fin de Grado TFG es una tarea académica que permite valorar las habilidades y competencias adquiridas por los estudiantes a lo largo de su periodo universitario. Así durante su desarrollo los estudiantes pueden poner en práctica numerosas bases teóricas de una forma amena y divertida. Desarrollar y controlar elementos robóticos de forma local o remota siempre ha supuesto un claro ejemplo de motivación adicional sobre los estudiantes.

Keywords—Brazo Robótico Articulado, Robot Móvil Autónomo, Reconocimiento de Gestos, Leap Motion Sensor, Microsoft Kinect, Asus Xtion Sensor.

I. INTRODUCCIÓN

Los grandes avances que ha alcanzado la robótica durante los últimos años, han hecho posible que esta disciplina de conocimiento haya pasado de considerarse una tecnología solamente al alcance de las grandes empresas o centros de investigación con presupuestos millonarios a estar prácticamente al alcance de todos. La aparición de aplicaciones de carácter robótico se ha multiplicado exponencialmente, de forma que es fácil percibir algunas de ellas en nuestra vida cotidiana. Esta serie de avances han provocado que la robótica se haya introducido también como complemento educativo en las enseñanzas básicas obligatorias [1]. Así la robótica proporciona un enfoque eficiente en el desarrollo de dispositivos asistidos debido a su gran funcionalidad y permite incrementar la calidad de vida.

La aparición de asignaturas como la Mecatrónica, en los actuales planes de estudio, ha permitido dotar de carácter multidisciplinar a la robótica e integrar y aglutinar a su vez varias disciplinas de conocimiento [2]. De modo que el estudiante durante el proceso educativo puede apreciar fácilmente las relaciones entre las diferentes asignaturas implicadas [3]. De esta forma se unen el aspecto mecánico,

eléctrico y electrónico junto con la estrategia de control para implementar toda la programación asociada [4].

En este documento se presenta una aplicación electrónica novedosa y de actualidad. El objetivo ha sido el desarrollo y construcción de un prototipo de brazo robótico articulado sobre una plataforma móvil y la implementación de una estrategia de control mediante el reconocimiento de gestos (Leap Motion Sensor) a través del movimiento natural del antebrazo y mano [5], ver figura 1. Se ha optado por el empleo de materiales reciclados y reutilizados con objeto de desarrollar un prototipo de bajo coste.

El brazo robótico opera en 3D y en su desarrollo se han analizado diversos sistemas de transmisión de movimiento, simulando los grados de libertad del antebrazo humano. La idea ha sido intentar que las acciones del sistema se parezcan lo máximo posible a su equivalente real. De este modo se han integrado varios sistemas y tecnologías en auge, como el desarrollo de un sistema de control basado en el dispositivo Leap Motion, la implementación del hardware constituido por μ P y la fabricación de piezas mediante tecnologías de impresión 3D, tecnologías que seguro serán de relevancia en los próximos años.



Figura 1. Varios ejemplos de dispositivos comerciales para desarrollar un control por reconocimiento de gestos. Microsoft Kinect, Asus Xtion Sensor y Leap Motion.

Mediante el control por gestos se ha pretendido favorecer la relación usuario-prototipo. El objetivo no es otro que mostrar al

estudiante la utilidad de la teoría con una situación real. Además de la estrategia de control por gestos se ha desarrollado, al mismo tiempo en paralelo, un sistema de control mediante un conjunto de pulsadores y una interface gráfica que permite al usuario interactuar con el sistema. Los objetivos iniciales se han resuelto de manera satisfactoria, obteniendo un prototipo que ha superado las expectativas iniciales y con un coste reducido.

II. STATE OF THE ART

El control por gestos es una tecnología de reciente aparición, donde los desarrolladores requieren de una gran capacidad de síntesis e interpretación de las necesidades del usuario, con objeto de modificar el manejo de aquellos dispositivos que utilizan métodos de control tradicionales. De este modo surge la oportunidad de experimentar con nuevos métodos de interacción que hasta el momento sólo existían en ciencia ficción, ver figura 2.

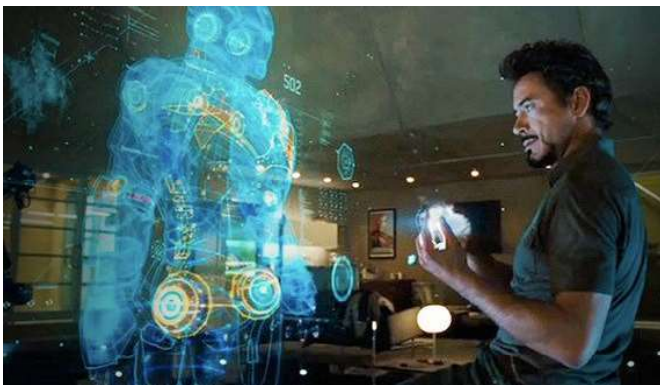


Figura 2. Fotograma de la película IronMan. En la imagen se muestra un interface inspirado en el concepto de manipulación de datos e información mediante gestos.

Con la implementación de los nuevos métodos de control, aparece la posibilidad de generar aplicaciones de carácter muy intuitivo. Como es lógico pensar, los gestos más eficaces serán aquellos en los que el usuario siente que el software se adapta a él y no al revés. En la vida cotidiana, el ser humano se mueve en la mayoría de las ocasiones por instinto, sin la necesidad de pensar antes de realizar el movimiento deseado. Existen diferentes dispositivos comerciales que permiten llevar a cabo una estrategia de control por gestos, ver figura 1. En la actualidad, esta serie de dispositivos son muy utilizados en videojuegos y otras aplicaciones interactivas destinadas al entretenimiento. Entre los más usuales podemos destacar: Leap Motion Controller, Microsoft Kinect y Asus Xtion Sensor (User Manual Reference [6-8]).

En estos últimos años numerosos autores han publicado estudios sobre la aparición e interacción con esta tecnología. Así El-laithy et al. [9] desarrolla un estudio para la implementación del sensor Microsoft X-Box Kinect destinado en aplicaciones de robótica. De este modo describe la tecnología de las cámaras RGB e infrarrojos que incorpora el dispositivo junto con su principio de funcionamiento básico. En este caso utiliza el sensor para observar la percepción en 3D del movimiento humano. Mientras González-Jorge et al. [10]

desarrolla una comparativa con respecto a la precisión que ofrecen los dispositivos Asus Xtion sensor y Microsoft Kinect para aplicaciones de metrología en entornos 3D. Por otro lado, Regazzoni et al. [11] analiza la eficiencia y limitaciones de diversas tecnologías de bajo coste, cámaras RGB y sensores RGB-D de profundidad, destinadas al reconocimiento de gestos y la captura de movimientos. El documento compara el comportamiento de dos aplicaciones comerciales basadas en sensores ópticos como Sony Playstation Eye y Microsoft Kinect.

A su vez, Bassily et al. [12] propone un nuevo algoritmo de control adaptativo destinado a un brazo articulado 6-DOF. El interface de comunicaciones hombre-máquina es desarrollado mediante el dispositivo Leap Motion Controller. Indica algunas características de otros dispositivos destinados a la interacción usuario-ordenador mediante una estrategia basada en el control por gestos, como por ejemplo: Nintendo Wii y Microsoft Kinect. Mientras Vargas et al. [13] presenta un sistema de reconocimiento de gestos centrado en el dispositivo Leap Motion Sensor y destinado a un prototipo de manipulación quirúrgica. De este modo describe la integración del sensor en la plataforma, las diferentes variables (roll, pitch, yaw) y los ajustes realizados.

Por otra parte, Ching-Hua et al. [14] junto con Mohandes et al. [15] y Elons et al. [16] utilizan el sensor 3D Leap Motion Controller y una metodología de control por gestos para el reconocimiento del lenguaje de signos. Hacen hincapié en su fácil programación debido a la incorporación de APIs (Application Programming Interfaces), la interface gráfica de usuario GUI (Graphical User Interface) y el bajo coste del dispositivo. Así pues describen el dispositivo Leap Motion y los algoritmos aplicados para la identificación y procesado de signos: modelo de Markov y redes neuronales. Por último Kobayashi et al. [17] propone un sistema de control de un brazo robótico basado en el reconocimiento de gestos. Utiliza un robot industrial y analiza las diferentes trayectorias seguidas durante el desplazamiento del brazo robótico. Mientras Lei et al. [18] estudia la similitud entre la posición de un humano y la imitación efectuada por un robot. El dispositivo comercial Microsoft Kinect es utilizado para el desarrollo de la aplicación. Los resultados experimentales demuestran una imitación satisfactoria y un rápido aprendizaje del robot utilizado, modelo Aldebaran Nao (25 DOF), [19].

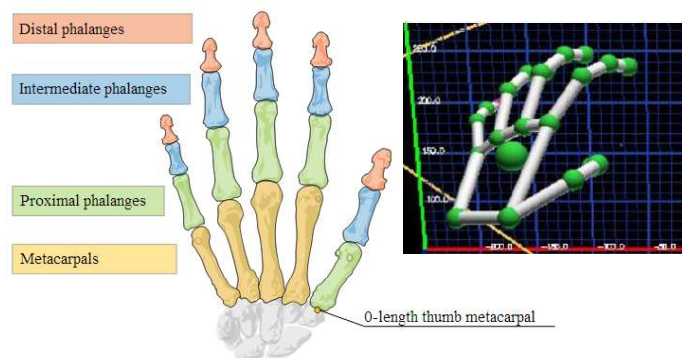


Figura 3. Número de puntos de referencia en la mano, representación 3D. El software analiza 22 puntos de control en total.

TABLA I. Comparación de dispositivos y sistemas para el reconocimiento de Gestos.

Dispositivo	Leap Motion Sensor	Wii Remote	Microsoft Kinect	MYO Gesture Control	Asus Xtion Sensor
					
Precio	71,50€	34,90€	147,07€	117,78€	177,97€
Precisión	Alta	Baja	Media	Media	Media
Arquitectura	2 cámaras monocromáticas 3 led IR	Acelerómetro led IR	cámara RGB cámara IR audio	Sensor EMG giroscopio acelerómetro	cámara RGB cámara IR audio
Procesador	MXIC MX25L3206E	Broadcom BCH2042	XCPU Xenon Xbox360	ARM Cortex M4	Intel X86 & AMD
Área de Operación	0,025-0,6m	0,8-6m	1,2-3,8m		0,35-3m
Complejidad de uso	Sencillo	Sencillo	Sencillo	Medio	Sencillo
Resolución			VGA (640*480)		SXGA (1280*1024)
Peso	45g	201g	1,3kg	93g	170g
Dimensiones	7,6*2,5*1,3cm	14,6*3,5*3,1cm	24,9*6,6*6,7cm	Diam. 10cm	18*3,5*5cm

III. LEAP MOTION CONTROLLER

Desde el lanzamiento al mercado del dispositivo Lead Motion una gran variedad de aplicaciones han sido desarrolladas, siendo las más populares las que permiten el control del ordenador a través de los movimientos de la mano. Sin embargo ninguna de ellas ha alcanzado un elevado grado de popularidad entre el usuario medio.

El principal objetivo en el desarrollo de la aplicación ha sido intentar que las acciones del brazo robótico planteado se parezcan lo más posible a su equivalente real. Pero siempre teniendo en consideración la influencia que el movimiento realizado provocará sobre el sistema. Esto es posible lograrlo mediante el escalado apropiado de los movimientos. Así el prototipo de brazo articulado construido tiene cuatro movimientos posibles, generados por distintos servomotores.

El reconocimiento de los diferentes gestos de la mano es desarrollado a través del dispositivo Leap Motion, un periférico provisto de sensores infrarrojos IR que reconoce el movimiento del antebrazo, mano y dedos simultáneamente, ver figura 3. Estas variables son procesadas por el ordenador con objeto de su descomposición en movimientos simples y básicos. Posteriormente toda esta información es enviada al brazo articulado. El control es realizado por medio del controlador Arduino, quién recibe los diferentes comandos procedentes del ordenador vía comunicación serie y transmite la información a cada uno de los servomotores implicados en el movimiento.

El prototipo de brazo articulado se ha implementado sobre un chasis con capacidad de movimiento, de modo que puede ser manejado por el usuario de manera remota mediante el sistema de comunicaciones inalámbrico del robot móvil. El objetivo ha sido desarrollar un brazo articulado mediante una estrategia de control por gestos que pudiese ser totalmente autónomo y sin cables.

A. Comparativa de Dispositivos de Control por Gestos

En la actualidad existen diferentes opciones comerciales en el mercado para desarrollar una estrategia de control por gestos, por ejemplo: Microsoft Kinect, Asus Xtion Sensor, Wii Remote, MYO Gesture Control o Leap Motion Controller, que es el dispositivo que finalmente ha sido seleccionado. La Tabla I recoge una breve comparación de las especificaciones entre los diferentes dispositivos mencionados.

Wii Remote. Este sistema está basado en un mando y una barra de sensores. Posee la capacidad de detección del movimiento en el espacio y la habilidad de seleccionar objetos en la pantalla. Fue creado para ser utilizado con una sola mano de manera lo más intuitiva posible. Mediante los sensores ópticos y el acelerómetro es posible conocer tanto la posición, velocidad o aceleración a lo largo de sus tres ejes.

Microsoft Kinect. Este dispositivo tiene unas características similares a Leap Motion. El dispositivo permite a los usuarios controlar e interactuar con la consola sin necesidad de tener contacto físico con el controlador tradicional (mando). El sistema dispone de un interface natural de usuario que permite reconocer gestos y comandos de voz. Entre sus características destaca la cámara RGB, un sensor de profundidad y un micrófono, ver figura 1. La principal ventaja del dispositivo es su gran área de reconocimiento, proporciona captura de movimiento de todo el cuerpo en 3D.

Asus Xtion Sensor. El dispositivo fue comercializado para el desarrollo de juegos y aplicaciones interactivas. Incorpora un sensor de movimiento que permite la detección de gestos y la captura completa del cuerpo humano en tiempo real. Así incluye una cámara RGB, un sensor de infrarrojos, detector de profundidad y un micrófono, ver figura 1.

MYO Gesture Control. Es un dispositivo electrónico, con forma de brazalete, para el reconocimiento de gestos. Permite captar la energía generada por los músculos y transformarla en

información. De este modo incorpora un conjunto de sensores electromiográficos (EMG) que detectan la actividad eléctrica del antebrazo, en combinación con un giroscopio y un acelerómetro para reconocer los diferentes movimientos.

B. Leap Motion Sensor

En el presente documento se ha optado por la elección del Leap Motion Sensor debido a que su grado de precisión es superior al resto de dispositivos comerciales [20]. Su reducida área de operación tampoco supone un problema ya que el control se desarrolla con el movimiento de mano y antebrazo (rango de movimientos admitidos por el dispositivo). Según su fabricante puede alcanzar los 200 frames por segundo mediante sus cámaras de infrarrojos y una precisión de hasta 0,2mm. Su campo de visión es de 150° con 0,25m³ aproximadamente de espacio 3D interactivo; en forma de pirámide invertida centrada sobre el dispositivo. Su alimentación junto con la transferencia de datos se efectúa a través del puerto USB.

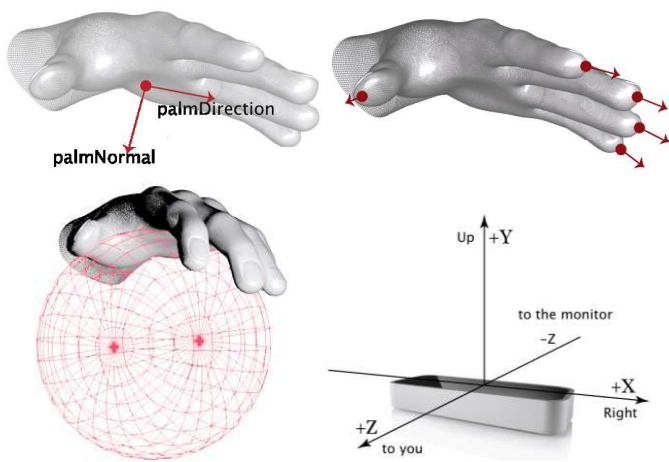


Figura 4. Sistema de ejes del dispositivo Leap Motion Sensor y análisis del desplazamiento de palma y dedos. También es posible apreciar la esfera virtual generada mediante la curva de la mano.

El dispositivo consta de dos cámaras monocromáticas junto con tres LEDs infrarrojos a través de los cuales genera un patrón 3D de puntos. De esta forma es capaz de obtener 200 imágenes por segundo de datos, que transfiere al ordenador para que el software desarrolle su posterior análisis mediante los algoritmos que integra.

El procesamiento de información se puede realizar mediante su SDK (Software Development Kit) que es descargado de la web <https://developer.leapmotion.com/>. Este software permite transformar las señales enviadas por el sensor Leap Motion en datos y vectores intuitivos como dedos, manos, posiciones o gestos, con objeto de facilitar el trabajo del programador, ver la tabla II. Así el SDK permite desarrollar aplicaciones APIs (Application Programming Interface) en una amplia variedad de lenguajes de programación como C++, Java, Python, JavaScript, Unity,... siendo posible descargar múltiples recursos de aplicación dentro de la comunidad de usuarios.

El software Leap Motion detecta el movimiento de la mano y dedos, ver figura 4, observando las posiciones en 3D. El software analiza el movimiento teniendo en consideración el frame anterior y el actual, sintetizando así los movimientos de

traslación, rotación y factores de escalado de cada uno de los objetos. Como variables considera los siguientes datos: número de manos y dedos, posición de cada mano y dedo, ángulo de la mano y velocidad de la palma. En la figura 4 se observa el sistema de coordenadas utilizado por el dispositivo.

El controlador contiene las APIs soportadas por el fabricante. A través del software, los datos de manos y dedos pueden ser enviados a los programas diseñados por el usuario. El objetivo es utilizar el dispositivo como un interface alternativo hombre-máquina. Actualmente han aparecido en el mercado muchas aplicaciones desarrolladas por medio de este controlador. En la mayor parte de los casos son juegos de entretenimiento o creación musical.

TABLA II. Parámetros obtenidos desde el software Leap Motion.

Palma de la Mano		Dedos	
Variable	Tipo	Variable	Tipo
Normal	vector	Direction	vector
Position	vector	Length	mm
Velocity	mm/sec	Tip position	vector
Confidence	float [0, 1]	Tip velocity	mm/sec
Pinch strength	float [0, 1]	Dip position	vector
Grab strength	float [0, 1]	Pip position	vector
Sphere center	vector	Mcp position	vector
Sphere radius	mm		

La tabla II enumera los principales parámetros de manos y dedos proporcionados por la API. Entre las características más sobresalientes relacionadas con la palma se incluyen: la normal (un vector unitario de dirección perpendicular a la mano), posición (la posición central de la palma de la mano) y velocidad (en milímetros por segundo). La fuerza de agarre (Grab strength) es un parámetro flotante entre cero y uno. Con "0" se indica la mano abierta mientras que "1" implica la mano cerrada. De forma similar, el parámetro Pinch strength muestra el rango de apertura entre el pulgar y otro dedo cualquiera de la misma mano. El centro de la esfera (Sphere center) y el radio (Sphere radius) se calculan sobre la base de una esfera virtual, suponiendo que la mano estuviese sosteniendo una pelota, ver figura 4.

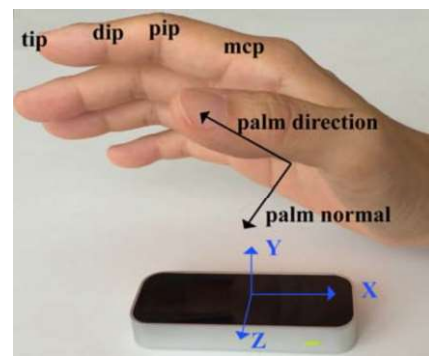








Figura 5.- Conjunto de variables y coordenadas proporcionadas por el software Leap Motion Sensor.

También es posible recoger la información y los parámetros correspondientes a los diferentes dedos. En este supuesto el software proporciona para cada dedo: la dirección (un vector unitario de dirección longitudinal) junto con su longitud (en milímetros), ver figura 4. Las posiciones entre las falanges (tip, dip y pip, ver figura 5) y metacarpianos (mcp) también son

TABLA III. Comparación de diferentes tarjetas controladoras embebidas.

Tarjeta Embebida	Arduino	Raspberry-Pi	Beagle Bone	Odroid XU4	HummingBoard	Udoo Dual
						
Precio	20€	30€	40€	74€	60€	100€
Tamaño	68,6*53,4mm	85*56mm	86,4*54,6mm	82*58mm	102*69mm	110*85mm
Arquitectura Procesador	Atmega328	Bradcom ARM11	TI Sitara ARM Cortex-A8	Samsung Exynos ARM Cortex-A15	Freescale ARM Cortex-A9	Freescale ARM Cortex-A9
Cores CPU	1	1	1	4	2	2
Velocidad	16MHz	700MHz	1GHz	2GHz	1GHz	1GHz
Memoria RAM	32kb	256Mb	512Mb	2Gb	1Gb	1Gb
Flash slots	No	microSD	microSD	microSD	microSD	microSD
Port USB	Si	Si	Si	Si	Si	Si
Port HDMI	No	Si	Si	Si	Si	Si
Ethernet	No	Si	Si	Si	Si	No
Audio	No	Si	HDMI	HDMI	Si	HDMI
Comunicación	I2C, SPI	I2C, SPI	I2C, SPI	I2C, SPI	I2C, SPI	I2C, SPI
Port I/O Digital	14	40	2*46	42	26	76
Otros Periféricos Genéricos	22 GPIO 10bits ADC PWM	17 GPIO UART	66 GPIO 12bits ADC CAN UART	GPIO ADC UART	8 GPIO CIR, FlexCAN UART	76 GPIO 10bits ADC PWM

registradas. Algunos de los parámetros proporcionados por la API no son adecuados para el control del brazo robótico.

Para obtener un dato relativo a la posición de la mano o dedos es necesario hacer referencia al objeto y escoger el dato que se desea procesar. Así a través de los siguientes comandos se obtiene:

Posición dedo:

```
frame.fingers(referencia_dedo).tipPosition().get(referencia_eje)
```

Posición palma:

```
frame.hands().palmPosition().get(referencia_eje)
```

Uno de los recursos incluidos dentro del software es Leap Motion Visualizer. Se trata de una aplicación donde es posible visualizar la representación 3D a través de la interacción con el dispositivo Leap Motion. Además este visualizador proporciona, en todo momento, información relativa al estado de las variables y parámetros de interacción, ver figura 6.

IV. TARJETA DE CONTROL EMBEBIDO

En la actualidad el mercado de hardware libre se ha convertido en un área en expansión. Hoy en día existen multitud de tarjetas microcontroladoras que debido a su bajo coste y versatilidad están siendo utilizadas tanto en robótica, mecatrónica como en investigación. Existe un gran abanico de tarjetas de hardware libre para la construcción de prototipos educativos como: Raspberry-Pi, Beagle Bone, Arduino, ODroid, Udoo Dual, Humming Board, CubieBoard, Rabbit Embedded, etc. Como característica más relevante destaca la conectividad ya que disponen de un gran abanico de puertos de

salida, lo que permite el acceso a diferentes tipos de periféricos. Así en la Tabla III se aprecia una breve comparativa de las diferentes especificaciones de algunas tarjetas controladoras comerciales.

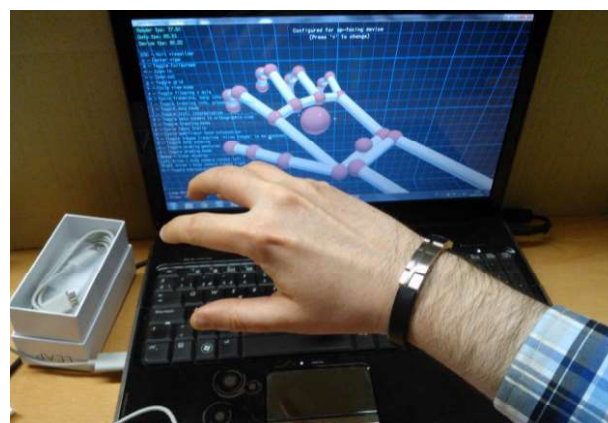


Figura 6. Ejemplo del entorno Leap Motion Visualizer destinado a la estrategia de reconocimiento de gestos.

Para el desarrollo del prototipo fue seleccionado el modelo Arduino Uno, ya que aunque se trata de uno de los modelos más básicos y sencillos cumple con los requerimientos y exigencias planteados. Es un tipo de controlador de código abierto y software de programación libre, muy versátil y asequible. De este modo la comunicación serie entre plataformas y el tratamiento de los datos procedentes del software de reconocimiento de gestos Leap Motion se han desarrollado a través de Processing. Se trata de un lenguaje de

programación y entorno de desarrollo integrado de código abierto basado en Java. Su entorno de trabajo es muy sencillo y manejable y su complejidad es relativamente baja; permitiendo la programación en Java y Python. La interacción entre Processing y Leap Motion se lleva a cabo por medio de una biblioteca, que nos permite manejar los datos adquiridos por el dispositivo.

V. DESCRIPCIÓN DEL BRAZO ARTICULADO

El brazo articulado fue dimensionado para ubicarlo sobre el chasis de un robot móvil autónomo. Uno de los objetivos ha sido el reciclaje de materiales en desuso. Así gran parte de las piezas móviles empleadas (ejes, engranajes, etc.) fueron extraídas de equipos obsoletos como impresoras. De esta forma el diseño inicial se ha adaptado a sus dimensiones. En el caso de las piezas más complejas que debían adaptarse a ejes, rodamientos o engranajes ya predefinidos, fueron fabricadas en plástico PLA y ABS por medio de una impresora 3D. La filosofía ha sido reducir al mínimo el coste de todo el sistema, ver figura 7.

De este modo el prototipo de brazo robótico diseñado permite los siguientes movimientos:

- Abrir/Cerrar pinza para coger objetos, mvto. “Pinza”.
- Giro sobre eje paralelo al suelo, mvto. “Muñeca”.
- Brazo telescópico extensible sobre plano horizontal paralelo al suelo, mvto. “Alargar”.
- Giro sobre eje perpendicular al suelo, mvto. “Giro”.

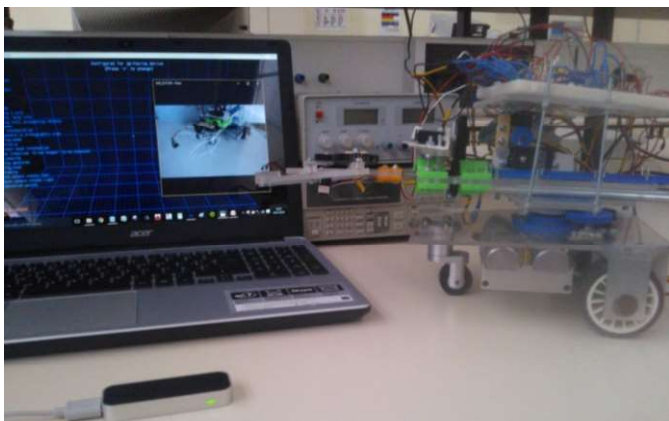


Figura 7. Prototipo de Brazo Robótico e interface de usuario. Implementación del sistema de control por gestos Leap Motion sobre el robot móvil autónomo.

En cada una de las articulaciones un servomotor HITEC permite desarrollar los movimientos de rotación o translación. A su vez los servomotores son controlados mediante una señal PWM. Algunos servomotores fueron modificados para ampliar su rango de movimiento, generándose la posibilidad de rotación de forma continua. Para ello se ha variado el control de posición interno del servomotor original. En este sentido para controlar la posición del elemento se ha generado un sistema de control en bucle cerrado basado en un encoder rotatorio externo (modelo DPL12 fabricado por Tyco Electronics). También se han incorporado microswitches a modo de finales de carrera para indicar el máximo límite del movimiento angular o lineal.

A. Sistema de transmisión Giro

A modo de ejemplo se muestra el movimiento de giro del brazo articulado sobre un eje perpendicular al suelo. Para este movimiento se optó por un sistema de engranajes como se muestra en la figura 8. Además fue introducida una rueda dentada auxiliar solidaria al eje del encoder rotatorio (engranaje de color azul) con objeto de conocer su posición angular en todo momento. A continuación se muestra la parte más destacable del código utilizado por Leap Motion para la identificación del movimiento y posición de la mano. Su finalidad es la realización de la función Giro.

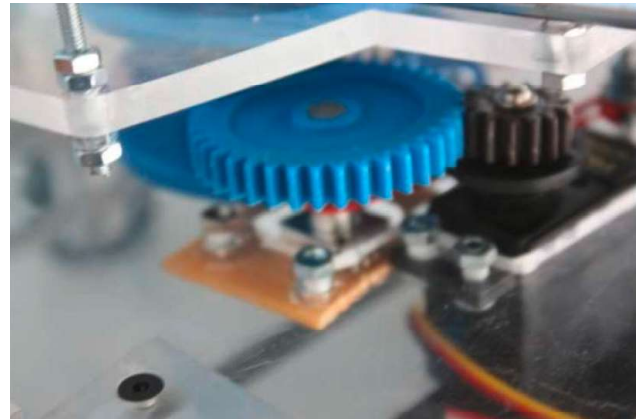


Figura 8. Detalle del sistema de rotación del brazo articulado (movimiento Giro). En la imagen se aprecia el servomotor modificado junto con el engranaje auxiliar solidario al encoder para obtener la posición angular.

```
//Función GIRO

int giro(final Controller controller){

    int giro=0;
    float posicionx = 0;
    if (controller.isConnected()){
        Frame frame = controller.frame();
        if (!frame.hands().isEmpty()){
            for (Hand hand : frame.hands()){

                //Toma de datos
                posicionx= hand.palmPosition().get(0);

                //Cálculo de posición
                giro=(int)(180*(posicionx+30)/(60));

                if (giro<=0){giro=0;}
                if (giro>180){giro=180;}

            }} return giro;}

```

Este movimiento se genera a través de la lectura de la posición de la palma de la mano sobre el eje X, ver ejes de coordenadas de la figura 4. El objetivo es generar una señal PWM que se transmite al servomotor HITEC consiguiendo que gire en un sentido u otro. El sistema fue planteado para que el brazo robótico pueda girar ± 90 grados desde la posición central de referencia. Esto conlleva varias vueltas completas del servomotor encargado del desplazamiento. En el esquema de la figura 9 se puede observar el sistema de transmisión de movimiento; representando los puntos de conexión de unos

subsistemas y otros. El chasis del robot posee dos ruedas motrices que permiten el desplazamiento y su orientación (movimiento tipo tanque) e incorpora una rueda loca en su parte delantera que sirve como elemento de apoyo.

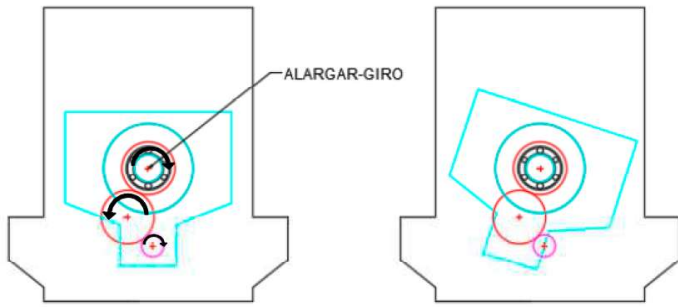


Figura 9. Esquema del movimiento Giro en el Brazo Articulado con respecto a la plataforma del Robot Móvil.

El engranaje solidario a la plataforma de giro posee $z_3 = 33$ dientes, mientras que el engranaje del encoder $z_2 = 38$ dientes y el engranaje del servomotor contiene $z_1 = 15$ dientes. Todos los engranajes tienen módulo $m = 1$ ($m = D_p/z_n$; ratio del diámetro primitivo del engranaje D_p y el número de dientes z_n). De este modo la relación de transmisión motor-base móvil “ i_{SB} ” es:

$$i_{SB} = \frac{z_2 z_3}{z_1 z_2} = 2,2 \quad (1)$$



Figura 10. Detalle del sistema de giro del soporte de la pinza en el brazo articulado (movimiento muñeca). La imagen muestra los dos engranajes solidarios al servomotor y al eje.

Como consecuencia de esta relación de transmisión i_{SB} , un giro de ± 90 grados desde la posición de referencia (variación angular $\Delta_G = 180^\circ$) conlleva 1,1 vueltas del accionamiento. Los valores de velocidad y par del servomotor utilizado (datos proporcionados por el fabricante) sin carga y con una alimentación de $V_{DD} = +5V$ son: velocidad $v_1 = 260^\circ/s$ y par $T_1 = 3,17kg \times cm$. Luego los parámetros máximos (v_B ; T_B) que es posible transmitir de manera ideal a la plataforma (base móvil) se corresponden con,

$$v_B = \frac{v_1}{i_{SB}} = 118,18^\circ/s \quad (2)$$

$$T_B = T_1 \times i_{SB} = 6,972kg \times cm \quad (3)$$

El sistema de control utiliza el encoder rotatorio DPL12 como variable de realimentación. Este dispositivo es del tipo incremental y proporciona 24 pulsos por vuelta. La relación de transmisión encoder-base móvil “ i_{EB} ” será

$$i_{EB} = \frac{z_3}{z_2} = 0,8684 \quad (4)$$

luego para el desplazamiento de ± 90 grados ($\pm 0,25$ vuelta) en la base móvil, el encoder proporciona un número de pulsos correspondiente con

$$N^\circ \text{ pulsos} = \pm 0,25 \text{ rev} \times i_{EB} \times 24 \frac{\text{pulse}}{\text{rev}} = \pm 5,2105 \quad (5)$$

Por sencillez del algoritmo de control, se adopta el valor entero superior ± 6 pulsos, lo que conlleva un giro máximo de $\pm 103,63$ grados en la base móvil del brazo articulado con respecto al punto de referencia (plataforma robot).

B. Sistema de transmisión Muñeca

Para el movimiento de muñeca se ha elegido un sistema constituido por dos engranajes ($m = 1$) del mismo diámetro. De este modo resulta sencillo transmitir el movimiento de 180° del servomotor hasta el eje sobre el cuál gira la pinza del robot. Este eje se apoya sobre un rodamiento que le permite girar con relativa facilidad, ver figura 10.

A continuación se muestra la parte más destacable del código implementado para la identificación del movimiento y giro de la mano. Su finalidad es la realización de la función Muñeca. Este movimiento se genera a través de la lectura de la posición relativa de los dedos índice y pulgar, considerando las variables en el eje Y.

```
//Función MUNECA
int muneca(final Controller controller){

int muneca=0;
float pospulgar1=0;
float posindice1=0;

if (controller.isConnected()){
Frame frame = controller.frame();
if (!frame.hands().isEmpty()){

//Toma de datos
Finger pulgar = frame.fingers().get(0);
pospulgar1 = pulgar.tipPosition().get(1);
Finger indice = frame.fingers().get(1);
posindice1 = indice.tipPosition().get(1);

//Cálculo de posición
muneca=(int)(180*(pospulgar1-posindice1+10)/(40));

if (muneca<=0){muneca=0;}
if (muneca>180){muneca=180;}

}} return muneca;}
```

En el diagrama de la figura 11 se puede observar el sistema de transmisión de giro de la pinza (muñeca); representando los puntos de conexión de unos subsistemas y otros. Los valores máximos de velocidad y par del servomotor HITEC HS-322 sin carga son los siguientes: par $T_2 = 3,17\text{kg}\times\text{cm}$, velocidad giro $v_2 = 260^\circ/\text{s}$.

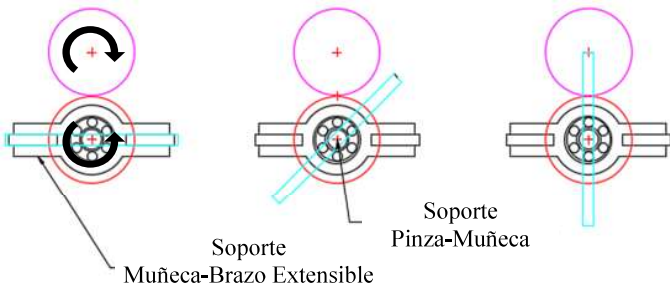


Figura 11. Esquema simplificado del movimiento Muñeca del Brazo Articulado sobre un eje horizontal paralelo al suelo.

C. Sistema de desplazamiento horizontal

En el desplazamiento horizontal del brazo articulado se ha utilizado un sistema piñón-cremallera con una rueda dentada que gira solidaria al servomotor. La cremallera se encuentra ubicada en la parte que se desplaza, mientras que el servomotor permanece anclado sobre la base de giro, ver figura 12. Un encoder rotatorio de tipo incremental DPL12 es utilizado como parámetro de control. Este movimiento se desarrolla mediante la lectura de la posición de la palma de la mano en el eje z.

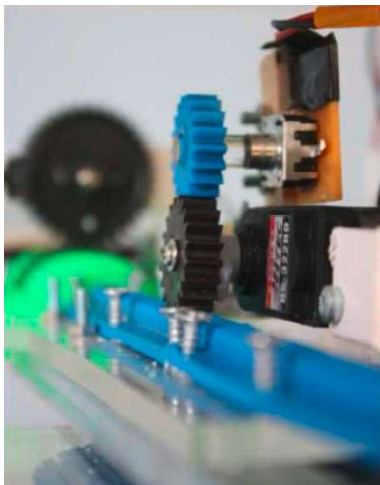


Figura 12. Detalle del sistema de desplazamiento horizontal. La imagen muestra los dos engranajes solidarios al servomotor y al brazo.

El engranaje solidario al servomotor contiene $z_4 = 20$ dientes mientras que el engranaje del encoder posee $z_5 = 18$ dientes. La cremallera tiene una longitud $l_C = 125\text{mm}$ (máximo desplazamiento horizontal), con una relación de 1 diente cada $2,5\text{mm}$, $a_C = 2,5\text{mm/diente}$. De este modo el ratio de transmisión servomotor-encoder " i_{EH} " será:

$$i_{EH} = \frac{z_4}{z_5} = 1,11 \quad (6)$$

Como consecuencia recorrer la longitud horizontal desde la posición de referencia ($\Delta l_C = 125\text{mm}$) conlleva 2,5 vueltas del servomotor. Los valores de velocidad y par del servomotor utilizado (datos proporcionados por el fabricante) sin carga y con una alimentación de $V_{DD} = +5\text{V}$ son: velocidad $v_3 = 260^\circ/\text{s}$ y par $T_3 = 3,17\text{kg}\times\text{cm}$. Luego la máxima velocidad de desplazamiento horizontal que posee el brazo articulado se corresponden con,

$$v_H = v_3 \frac{z_4}{360^\circ} a_C = 36,11\text{mm/s} \quad (7)$$

El sistema de control utiliza el encoder rotatorio DPL12 como variable de realimentación. Este dispositivo es del tipo incremental y proporciona 24 pulsos por vuelta. Así el número de pulsos que proporciona el encoder para recorrer la longitud l_C se corresponde con

$$N^\circ \text{ pulsos} = \frac{l_C}{a_C z_4} \times i_{EH} \times 24 \frac{\text{pulse}}{\text{rev}} = 66,664 \text{ pulsos} \quad (8)$$

De este modo el encoder proporciona 1 pulso cada $1,875\text{mm}$ de recorrido horizontal ($k_C = 1,875\text{mm/pulso}$). Este parámetro es una constante a incorporar en el sistema de realimentación.

```
//Función ALARGAR
int alargar(final Controller controller){
    int alargar=0;
    float posiciony = 0;
    if (controller.isConnected()){
        Frame frame = controller.frame();
        if (!frame.hands().isEmpty()){
            for (Hand hand : frame.hands()){
                //Toma de datos
                posiciony= hand.palmPosition().get(2);
                //Cálculo de posición
                alargar=(int)(180*(-posiciony+60)/(120));
                if (alargar<=0){alargar=0;}
                if (alargar>180){alargar=180;}
            }}return alargar;}
```

VI. CONCLUSIONES

La realización de un TFG aplicado, en el ámbito de las ingenierías, permite la evaluación de habilidades, destrezas y competencias adquiridas por los estudiantes. Del mismo modo hace posible incrementar el grado de motivación durante su realización. En este documento se ha presentado el diseño y construcción de un prototipo de brazo robótico articulado destinado al chasis de un robot móvil autónomo. La filosofía de trabajo ha sido la reutilización y reciclaje de materiales provenientes de equipos obsoletos y ya en desuso, lo que ha supuesto una disminución del coste del prototipo.

La incorporación de una estrategia de control mediante Leap Motion, basada en el reconocimiento de gestos, ha permitido favorecer la relación con el interface usuario-prototipo. Además ha permitido dotar al usuario de una mayor capacidad de control y facilitar su manejo. De este modo, se ha desarrollado un prototipo de brazo robótico capaz de operar en 3D simulando los grados de libertad (DOF) del antebrazo humano. Los objetivos iniciales se han resuelto de manera satisfactoria, obteniendo como resultado un prototipo que ha superado las expectativas iniciales y con un coste reducido.

AGRADECIMIENTOS

Los autores agradecen la ayuda financiera concedida por la Universidad de Zaragoza para el desarrollo del presente trabajo, a través del Programa de Innovación Docente para grupos de profesores (PIIDUZ). Convocatoria 2015/16, línea 1: Aplicar metodologías activas que supongan una mejora del proceso enseñanza-aprendizaje, desarrollar experiencias de innovación docente apoyadas en las nuevas tecnologías, con el objetivo de mejorar el diseño curricular de las asignaturas y mejorar las metodologías y recursos asociados. Referencia PIIDUZ_15_019.

REFERENCIAS

- [1] Artal J.S., Caraballo J. and Dufo R. "DTMF technology applied to the identification and control of a small mobile robot". Technologies Applied to Electronics Teaching (TAEE 2014). IEEEExplore Digital Library. 11-13 June 2014. Bilbao (Spain); pp. 01-06.
- [2] Valera A., Soriano A. and Vallés M. "Low-Cost Platforms for Realization of Mechatronics and Robotics Practical Works". Revista Iberoamericana de Automática e Informática Industrial RIAI. Elsevier ScienceDirect. Volume 11, Issue 4, October-December 2014, Pages 363-376.
- [3] Artal J.S., Caraballo J. and Bandres R. "Incorporation of mobile phones in small robotics applications like educational tool". Technologies Applied to Electronics Teaching (TAEE 2012). IEEEExplore Digital Library. 13-15 June 2015. Vigo (Spain); pp. 181-186.
- [4] Skraba A., Kolozvari A., Kofjac D. and Stojanovic R. "Wheelchair maneuvering using leap motion controller and cloud based speech control: Prototype realization". 4th Mediterranean Conference on Embedded Computing (MECO2015). IEEEExplore ScienceDirect. Page(s): 391-394.
- [5] Yao D., Jiang M., Abulizi A. and You X. "Decision-tree-based algorithm for 3D sign classification". 12th International Conference on Signal Processing (ICSP2014). IEEEExplore ScienceDirect. 19-23 October 2014. Hangzhou, Pages: 1200 - 1204.
- [6] Reference manual Leap Motion Sensor. (February 2016). <http://www.leapmotion.com>. Leap motion's Developer Guide.
- [7] Microsoft Kinect for Windows Programming Guide. (December 2015). <https://msdn.microsoft.com/en-us/library/hh855348.aspx>. Architecture Kinect Sensor.
- [8] Manual Asus Xtion Sensor. Developer Guide. (December 2015). https://www.asus.com/3D-Sensor/Xtion_PRO/
- [9] El-laithy R.A., Huang J. and Yeh M. "Study on the use of Microsoft Kinect for robotics applications". IEEE/ION Position Location and Navigation Symposium (PLANS2012). IEEEExplore Digital Library. 23-26 April 2012. Myrtle Beach, SC. Pages: 1280 - 1288
- [10] Gonzalez-Jorge H., Riveiro B., Vazquez-Fernandez E., Martínez-Sánchez J. and Arias P. "Metrological evaluation of Microsoft Kinect and Asus Xtion sensors". Measurement, Elsevier ScienceDirect. Volume 46, Issue 6, July 2013, Pages 1800 - 1806.
- [11] Regazzoni D., Vecchi G. and Rizzi C. "RGB cams vs RGB-D sensors: Low cost motion capture technologies performances and limitations". Journal of Manufacturing Systems. Elsevier ScienceDirect. Volume 33, Issue 4, October 2014, pp 719-728.
- [12] Bassily D., Georgoulas C., Guettler J., Linner T. and Bock T. "Intuitive and Adaptive Robotic Arm Manipulation using the Leap Motion Controller". 41st International Symposium on Robotics; Proceedings of ISR/Robotik 2014. IEEEExplore ScienceDirect. 2-3 June 2014. Munich, Germany. Pages: 1 - 7.
- [13] Vargas H.F. and Vivas O.A. "Gesture recognition system for surgical robot's manipulation". XIX Symposium on Image, Signal Processing and Artificial Vision (STSIVA2014). IEEEExplore ScienceDirect. 17-19 September. Armenia; pages 1 to 5.
- [14] Ching-Hua C., Regina E. and Guardino C. "American Sign Language Recognition Using Leap Motion Sensor". 13th International Conference on Machine Learning and Applications (ICMLA 2014). IEEEExplore ScienceDirect. 03-06 December. Detroit, pp 541 to 544.
- [15] Mohandes M., Aliyu S. and Deriche M. "Arabic sign language recognition using the leap motion controller". IEEE 23rd International Symposium on Industrial Electronics (ISIE2014). 1-4 June 2014. Istanbul. Pages: 960 - 965
- [16] Elons A.S., Ahmed M., Shedid H. and Tolba M.F. "Arabic sign language recognition using leap motion sensor". 9th International Conference on Computer Engineering & Systems (ICCES2014). IEEEExplore ScienceDirect. 22-23 December. Cairo. Pages: 368 - 373.
- [17] Kobayashi F., Okamoto K. and Kojima F. "Robot-human handover based on position and posture of human hand". 15th International Symposium on Soft Computing and Intelligent Systems (SCIS) and 7th International Conference on Advanced Intelligent Systems (ISIS), 2014. IEEEExplore ScienceDirect. Volume 33, Issue 4, 3-6 December 2014, Kitakyushu (Japón), pp 918 - 921.
- [18] Jie Lei, Mingli Song, Ze-Nian Li and Chun Chen. "Whole-body humanoid robot imitation with pose similarity evaluation". Signal Processing. Elsevier ScienceDirect. Volume 108, March 2015, pp 136 - 146.
- [19] Información obtenida de la web de robótica educativa <https://www.aldebaran.com/en/solutions/education-research> (April 2016).
- [20] Marin G., Dominio F. and Zanuttigh P. "Hand gesture recognition with leap motion and kinect devices". IEEE International Conference on Image Processing (ICIP 2014). IEEEExplore ScienceDirect. 27-30 October. Paris (France), pp 1565 to 1569.