



PROSPECTIVA

ISSN: 1692-8261

rprospectiva@gmail.com

Universidad Autónoma del Caribe

Colombia

Posada Aguilar, José David; Villanueva Padilla, Jair; Castillo Perez, Moises David; Molina
Gonzalez, Steven de Jesus

Control de un brazo robótico usando el hardware kinect® de microsoft

PROSPECTIVA, vol. 11, núm. 2, julio-diciembre, 2013, pp. 88-93

Universidad Autónoma del Caribe

Disponible en: <http://www.redalyc.org/articulo.oa?id=496250736011>

- Cómo citar el artículo
- Número completo
- Más información del artículo
- Página de la revista en redalyc.org

redalyc.org

Sistema de Información Científica

Red de Revistas Científicas de América Latina, el Caribe, España y Portugal

Proyecto académico sin fines de lucro, desarrollado bajo la iniciativa de acceso abierto

Control de un brazo robótico usando el hardware kinect® de microsoft

Robotic arm control based on microsoft kinect® hardware

José David Posada Aguilar¹, Jair Villanueva Padilla²,
Moises David Castillo Perez³, Steven de Jesus Molina Gonzalez³

¹ Profesor de Tiempo Completo, Universidad Autónoma del Caribe, Grupo de Investigación (IET-UAC), jdposa@gmail.com

² Profesor de Tiempo Completo, Universidad Autónoma del Caribe, Grupo de Bioingeniería, CEBI-UAC, jvillanueva@uac.edu.co

³ Estudiante de Ingeniería Electrónica y Telecomunicaciones, IET-UAC

Recibido 15/09/13, Aceptado 28/11/2013

RESUMEN

Los Joysticks y las interfaces hápticas han sido por excelencia los mecanismos más utilizados para controlar un brazo robótico manipulador. Sin embargo, en campos específicos como los de la cirugía robótica, es deseable que la interfaz sea de fácil uso, e intuitiva para el cirujano, tal que mejore sustancialmente las capacidades del mismo. En este artículo se presentan una estrategia para el control de los movimientos de un brazo robótico con dos grados de libertad utilizando como referencia dos movimientos de una mano. Para la utilización de la construcción del brazo robótico se utiliza la plataforma de Lego MindStorm NXT. Para la detección de las articulaciones se utilizó la librería de detección de articulaciones del SDK de Microsoft Kinect®. La implementación de un controlador para el robot, garantizó que la posición capturada de la mano sea la posición que presenta el brazo robótico. Por último, se validó el comportamiento del sistema mostrando el seguimiento del brazo robótico a la posición deseada obteniendo un bajo error cuadrático para el seguimiento de los dos ángulos requeridos.

Palabras clave: Detección de movimiento, Control de Posición, Lego Mindstorm®, Cirugía Robótica, Detección de Articulaciones.

ABSTRACT

The Joysticks and the haptic interfaces has been the most used mechanisms to control a robotic arm., However, at research fields such as the robotic surgery, the interface must be easy to use an intuitive for the surgeon. This article presents, a control strategy of a two degree of freedom robotic arm, using as reference the movements of a human hand detected by the Kinect® sensor, using the SDK libraries. The controller implementation guaranteed that the captured hand position would be the robotic arm position. The strategy was validated using a graph showing the actual and the proposed trajectory, and also the squared error of the system.

Keywords: Motion detection, Position Control, Lego Mindstorm®, Robotic Surgery, Joint Detection.

1. INTRODUCCIÓN

Desde sus inicios, la robótica ha contribuido significativamente en distintos campos, en el ámbito clínico, industrial e investigativo. En el contexto clínico, la robótica médica ha permitido realizar intervenciones quirúrgicas con un alto nivel de precisión y corte. Estos tipos de sistemas permiten reducir la probabilidad de muerte o error en una cirugía y al mismo disminuir el tiempo de intervención y los costos asistenciales.

Los principales exponentes de estos sistemas robóticos son el robot Da Vinci y el robot Zeus [1]. El primer sistema está

compuesto por una consola de mando, en la cual el cirujano visualiza el procedimiento a través de una cámara 3D y controla los brazos robóticos. El robot cuenta con tres brazos fijos y uno opcional para algunos procedimientos especializados. Estas partes se encuentran físicamente separadas, y conectadas por medio de cables los cuales transportan audio y video para facilitar el trabajo entre el médico y el personal asistencial. Las intervenciones más comunes realizadas por este sistema son: cirugías cardíacas, ginecológicas y urológicas [2].

El sistema robótico Zeus se encuentra conformado por subsistema separados físicamente, una consola de mando conectada a tres brazos robóticos los cuales se encuentran

a una mesa de operación. El brazo que carga el endoscopio es controlado por la voz y los otros dos brazos con cuatro grados de libertad sostienen los demás instrumentos los cuales son manipulados con un Joystick desde la consola del cirujano. El panorama quirúrgico es visualizado con una imagen de dos dimensiones con una pantalla estándar localizada en la misma consola o con unas gafas que permiten visualizar la imagen en 3D [3].

A nivel nacional se tienen reportes del uso de este tipo de sistemas en la ciudad de Bogotá, la cual, desde el año 2010, donde se han realizado diferentes procedimientos quirúrgicos tales como: histerectomías, prostatectomías radicales, miomectomías, entre otras [4].

Uno de los principales retos de este tipo de sistemas consiste en el desarrollo de interfaces eficientes entre el cirujano y el robot. En la actualidad, el joystick y las interfaces hápticas son los mecanismos más utilizados para operar un robot, no obstante se hace importante explorar las posibilidades de interfaces basadas en visión y voz [5].

En el ámbito Latinoamericano, se destaca el diseño y desarrollo de una interfaz humana con sensores articulares para controlar un robot antropomórfico de dos grados de libertad. El proyecto fue desarrollado por el grupo de Robótica de la Universidad Autónoma de Puebla, México (2009). La interfaz permite adquirir las señales provenientes de una extremidad superior a través de dos acelerómetros, transmitirlos a una tarjeta de adquisición de datos para su procesamiento y el control de los movimientos del brazo robótico. La plataforma utilizada para la visualización y el procesamiento de los datos fue Labview [6].

Así mismo, el Grupo de Robótica Médica de la Escuela de Ingeniería Industrial de la Universidad de Málaga (España), diseñó un robot de asistencia para cirugías laparoscópicas capaz de comprender las órdenes suministradas de un cirujano por medio de gestos y de la voz, permitiendo el manejo de la lámpara laparoscópica y los instrumentos quirúrgicos dentro de una sala de cirugía. Con este desarrollo se logró alcanzar un alto porcentaje de precisión y disminución en el tiempo de intervención en el paciente, consiguiendo que el 90% de las instrucciones dadas al robot a través de comandos por voz fueran reconocidas [7].

No obstante, en los últimos años ha surgido un gran interés en desarrollar interfaces usando el sensor Kinect® de Microsoft para controlar y operar robots. Es así como Departamento de Ingeniería Robótica y Ciencia de la Computación del Worcester Polytechnic Institute, implementa una estrategia para operar un robot humanoide utilizando imágenes de profundidad y un software de seguimiento del esqueleto para controlar la navegación, gestos faciales y movimientos de los brazos [8].

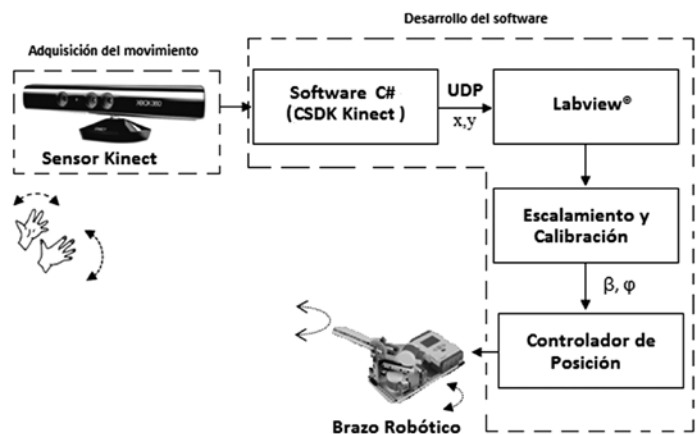
El propósito de este proyecto es el desarrollo de una estrategia para el control del movimiento de un brazo robótico con dos grados de libertad utilizando como interface el sensor Kinect® de Microsoft de Microsoft.

2. METODOLOGÍA

La metodología implementada para llevar a cabo este proyecto se observa en la figura 1.

Figura 1. Metodología implementada.

Figure 1. Methodology



A. Adquisición del movimiento de la mano

El dispositivo que se utiliza para este trabajo es el sensor Kinect® de Microsoft®, el cual incorpora un hardware avanzado de varios sensores. Un sensor de profundidad, una cámara de color RGB, y una matriz de cuatro micrófonos, que permite capturar el movimiento 3D de todo el cuerpo, realizar un reconocimiento facial, y el reconocimiento de voz. En la figura 2 se muestra el sensor Kinect® utilizado [9].

El proyector y la cámara RGB infrarroja, la cual tiene como objetivo fundamental proyectar un láser en el área de interés del sensor. Posterior a esto, captura la imagen y toma esta misma enviándola a un conjunto de filtros, para que el Kinect®, posteriormente determine si es una persona o no. Las librerías del Kinect® juegan un papel fundamental en este proyecto, debido a que generalmente esta herramienta, es utilizada para el desarrollo de software como juegos de video en donde se necesita el seguimiento de la trayectoria del cuerpo humano, donde se implementa estas bases, para el desarrollo del proyecto. El funcionamiento de las librerías utilizadas es descrito en la siguiente sección [9].

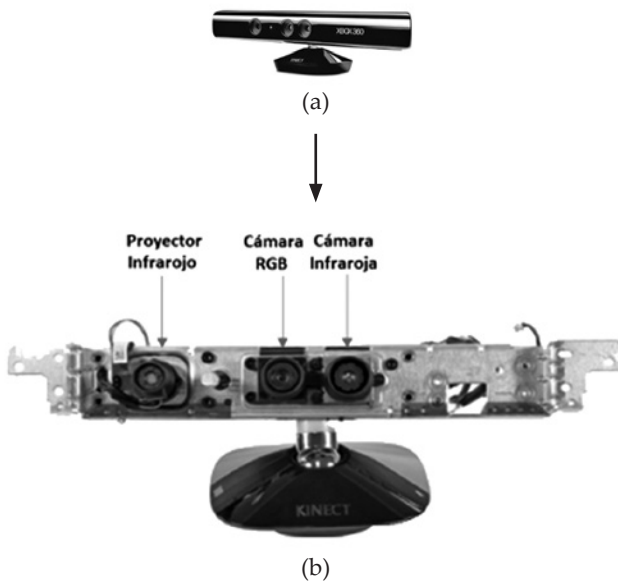
Así mismo, para la construcción del brazo robótico se utiliza la plataforma de Lego MindStorm NXT [10], la cual

está compuesta por bloques de construcción, servomotores y sensores. Los servomotores y los sensores son controlados por un bloque programable inteligente, el cual cuenta con un microprocesador de 32-bits, memoria Flash y conectividad Bluetooth, el cual puede ser programado utilizando un software de desarrollo como Labview. En la figura 3 se muestra el esquemático de brazo robótico construido y el prototipo desarrollado con los bloques del Lego MindStorm NXT. El diseño del brazo construido fue tomado de [11].

El ángulo β mapea la posición en x de la mano, mientras que el ángulo ψ mapea la posición en y . En la siguiente sección se describe el software de Labview para garantizar el control de cada uno de los ángulos.

Figura 2. Sensor del Kinect® de Microsoft. a) Sensor Kinect®. b) Proyector Infrarrojo (IR), Cámara a color (RGB) y cámara infrarroja (IR) dentro del Sensor del Kinect® [12].

Figure 2. (a) The Kinect® Sensor. (b) The infrared (IR) projector, IR camera, and RGB camera inside a Kinect® sensor.



B. Desarrollo del Software para aplicación robótica

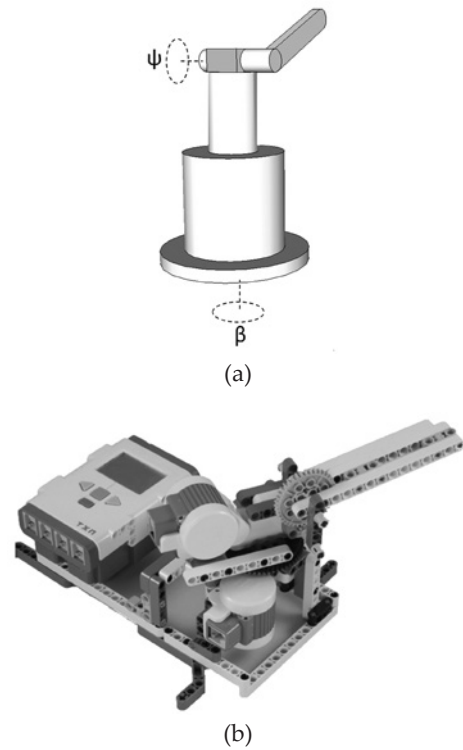
Para el desarrollo de la aplicación se utilizaron dos software de desarrollo: Visual Estudio C# en conjunto con el SDK (Software Development Kit, por sus siglas en inglés) y Labview con LEGO Mindstorms NXT y el PID and Fuzzy Logic Toolkit.

Con C# se desarrolló la aplicación que detecta los movimientos y la posición de la mano relativa a la posición del

Kinect®, esto se describe gráficamente en la figura 5, donde se presenta el espacio coordenado en donde se captura la posición de las articulaciones del cuerpo humano. En este trabajo solo se pretende detectar la posición en x , y de la mano derecha, descartando el resto de las articulaciones del cuerpo. En total la librería de detección de movimientos articulares incluida en el SDK del Kinect® tiene las capacidad de detectar hasta 20 articulaciones fundamentales del cuerpo humano, tal como se muestra en la figura 4, en la cual se muestra que la mano hace parte de este conjunto [13].

Figura 3. (a) Esquemático del brazo robótico construido, (b) Brazo robótico construido usando los bloques de Lego[11].

Figure 3. (a) Robotic arm schematic, (b) Robotic arm built using Lego blocks.



Una vez capturada la posición de la mano derecha, se transmite a través de un puerto independiente para cada eje coordenado utilizando el protocolo de comunicación UDP. Esta posición se recibe en una aplicación desarrollada en el software Labview, para posteriormente enviarla como un comando a los lazos de control para cada uno de las posiciones de los ángulos, tal como se presenta en la figura 6. Además, en la figura se muestra que es necesario realizar un escalamiento entre las coordenadas x , y capturadas y los ángulos requeridos. Como parámetros de dicho escalamiento, se encuentran las coordenadas en x , y máximas y mínimas, las cuales

dependen del entorno en el cual se realice la prueba, es decir a la distancia a la que la persona se encuentra del sensor y de la magnitud del movimiento que realiza. Los otros dos parámetros son los ángulos máximos y mínimos de desplazamiento de las articulaciones de brazo robótico, los cuales se determinan en función de la seguridad y el espacio a cubrir con el mismo. La sintonía del controlador se realiza utilizando el PID and Fuzzy logic Toolkit a través de un bloque de autosintonía.

Figura 4. Rango de captura para cada eje en el espacio de las juntas [13].

Figure 4. Capture range for each axis at joint space.

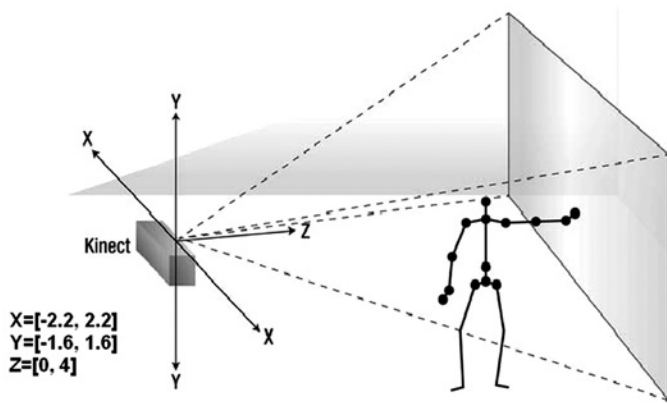


Figura 5. Descripción de las Juntas[13].

Figure 5. Joints description.

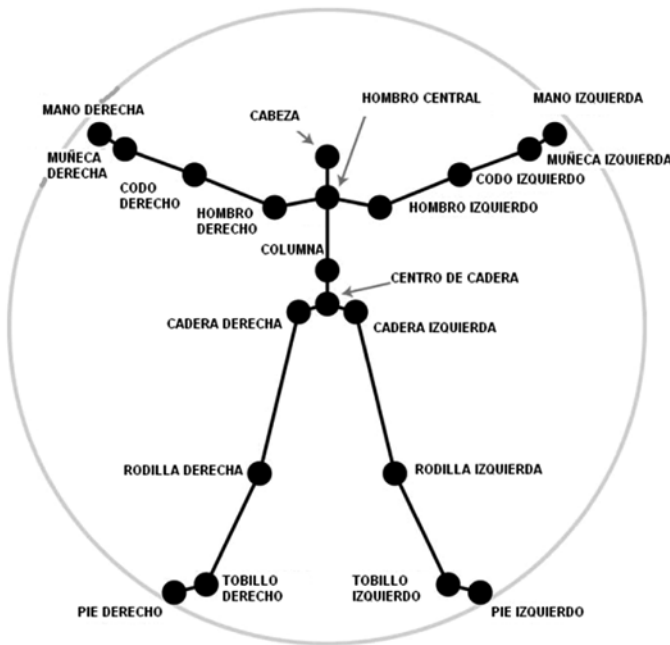
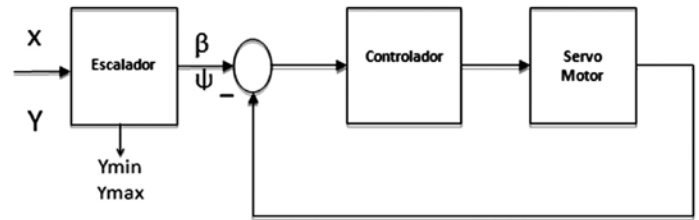


Figura 6. Lazo de Control de los servos motores.

Figure 6. Servo motor control loop



C. Evaluación del Prototipo

Para comprobar la efectividad de la metodología propuesta se procedió a configurar un escenario de prueba, en el cual el individuo moviese su mano derecha y se pudiese comprobar el seguimiento del brazo robótico a la trayectoria propuesta.

En primera instancia se realizó un procedimiento de ajuste inicial, en el cual consideraron los desplazamientos máximos y mínimos esperados para la mano derecha. Así mismo, se restringieron los ángulos mínimos y máximos de desplazamiento del brazo por razones de integridad del sistema. De la misma manera, se procedió a sintonizar los controladores de posición para los ángulos β y ψ . La ganancia de dichos controladores fue hallada utilizando el software de auto sintonización de Labview.

3. RESULTADOS

En la tabla 1 se presentan los valores para la configuración inicial del sistema. Para esta configuración se consideraron los criterios relacionados en la sección anterior. También se observa el resultado del procedimiento de autosintonía para un controlador proporcional observando sus ganancias.

Tabla 1. Resumen de los valores para la configuración inicial.

Table 1. Summary of initial configuration values

Parámetro	Valor
Valor mínimo de X	-0.3
Valor máximo de X	0.3
Valor mínimo de Y	-0.3
Valor máximo de Y	0.3
Valor mínimo de β	-100°
Valor máximo de β	100°
Valor mínimo de ψ	-100°
Valor máximo de ψ	100
Ganancia del controlador del ángulo β	3,48801
Ganancia del controlador del ángulo ψ	2,936899

Una vez hallado los valores iniciales necesarios para el funcionamiento del sistema se procedió a realizar la prueba con un individuo. Los resultados del seguimiento de la trayectoria propuesta por la mano derecha y ejecutada por el brazo para cada movimiento angular se muestran en la Figura 7 y 8.

Figura 7. Respuesta del seguimiento del sistema para el ángulo β

Figure 7. System response of β angle.

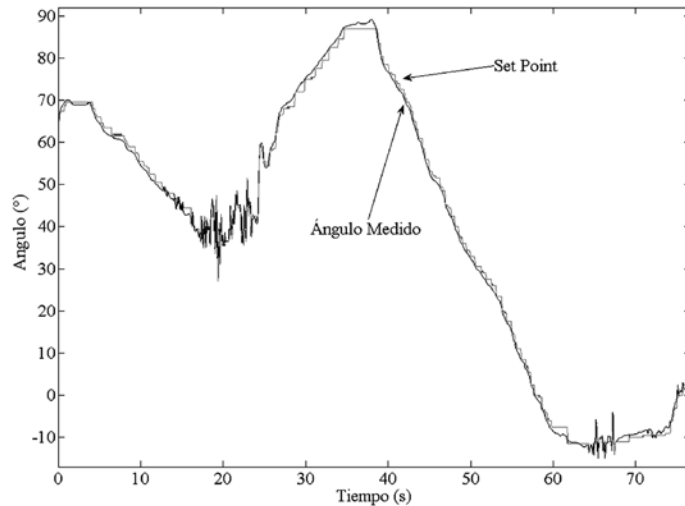
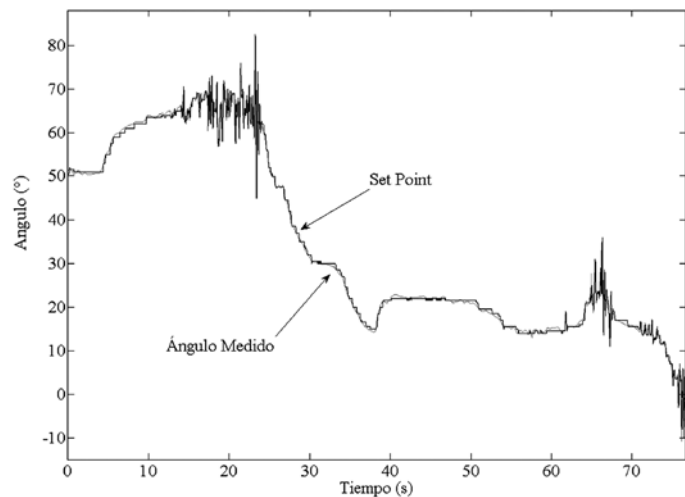


Figura 8. Respuesta del seguimiento del sistema para el ángulo ψ

Figure 8. System response of ψ angle.



En la figura 9 y 10 se evidencia detalladamente los errores de seguimiento para cada uno de los ángulos. Se observa que cuando los movimientos propuestos siguen una trayectoria suave el error tiende a ser cero. No obstante cuando los movimientos propuestos son abruptos, el sistema dada su velocidad limitada no respondió de igual forma.

Lo anterior, permite vislumbrar la posibilidad de probar la estrategia implementada con un brazo robótico más rápido y así mismo mejorar la forma como las aplicaciones de C# y Labview se comunican, de manera que se aumente la velocidad de respuesta del sistema.

Figura 9. Error de seguimiento para el ángulo β .

Figure 9. Error response for β angle.

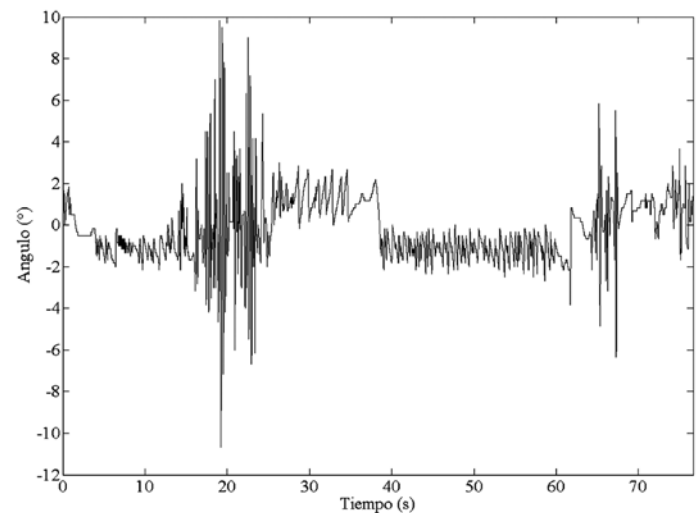
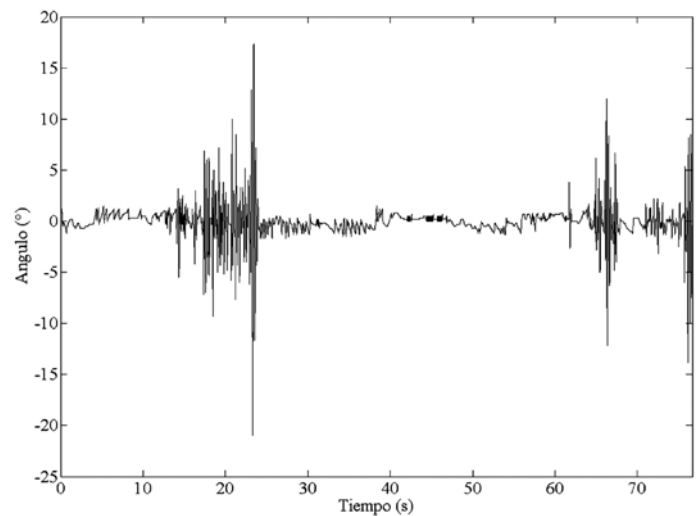


Figura 10. Error de seguimiento para el ángulo ψ .

Figure 10. Error response for ψ angle.



Finalmente, tal como se observa en la tabla 2 el error cuadrático medio para el ángulo β fue menor que para el ángulo ψ . Esto se debe principalmente a la inercia a la que se encontraba sometido dicho motor, la cual era mayor, dada la construcción del brazo. Una mayor inercia ayudó en los momentos en los cuales se ejecutaron movimientos bruscos de la mano en intervalos de tiempo cortos, tal que la respuesta del sistema no fuese tan rápida.

Los porcentajes de acierto para los ángulos β y ψ fueron calculados considerando el vector de datos disponible para el set point y las mediciones obtenidas comparándolas una a una con una tolerancia de 0.5° para determinar si el ángulo está siguiendo acertadamente o no, tal como se observa en la tabla 2, donde los porcentajes obtenidos fueron 98.88% y 97.6% para los ángulos β y ψ respectivamente

Tabla 2. Resumen de los valores para la configuración inicial.
Table 2. Summary of initial configuration values

Parámetro	Valor
Error cuadrático para el seguimiento del ángulo β	9.1075e3
Error cuadrático para el seguimiento del ángulo ψ	1.0515e4
Porcentaje de aciertos para el ángulo β	98.88%
Porcentaje de aciertos para el ángulo ψ	97.6%

CONCLUSIONES

- El desarrollo de soluciones que utilicen el sensor Kinect® de Microsoft® está en hoy en aumento, dada la facilidades que la compañía ha brindado a través del SDK, el cual es de libre uso para el campo académico. Es así como en esta investigación se usó el Kinect® como interfaz para el control de un brazo robótico, en sustituto al control tradicional, a través de un teclado o un joystick, permitiendo una interfaz más natural para el mismo, obteniendo unos resultados preliminares promisorios con aspectos a trabajar en investigaciones posteriores. Se pudo observar como el control de posición funcionó de forma eficiente, y la posición propuesta por la mano pudo ser seguida con un error menor a 5° la mayor parte del tiempo.
- Dentro de los aspectos futuros a trabajar están la mejora en la precisión de la captura de la posición de la mano, el uso de un brazo robótico con más precisión en sus movimientos y la planificación de la trayectoria en caso de que dos posiciones capturadas exijan un movimiento no adecuado por parte del robot o por parte de la tarea específica asignada.
- Lo anterior, permite vislumbrar la posibilidad de probar la estrategia implementada con un brazo robótico más rápido y así mismo mejorar la forma como las aplicaciones de C# y Labview se comunican, de manera que se aumente la velocidad de respuesta del sistema.

REFERENCIAS

- [1] Pugin, F., Bucher, P., Morel P., History of robotic surgery: From AESOP and Zeus to da Vinci. J. of Visceral Surgery, 148, e3-e8, 2011.
- [2] Surgery Enabled by Da Vinci. (2012). [Internet], Disponible desde: <<http://www.davincisurgery.com/davincisurgery/>> [Acceso 20 de noviembre 2012].
- [3] Kalan S., Chauhan S., Coelho RF. et al., History of Robotic Surgery, J.of Refractive Surgery,4(3),141-147, 2010.
- [4] La cirugía laparoscópica asistida por robot. (2012) [Internet], Disponible desde: http://www.marly.com.co/serv_cirugia_robotica.html>. [Acceso 20 de noviembre 2012].
- [5] Luo R., Perng Y. W., Advances of mechatronics and robotics, IEEE Ind. Elect. Mag., 27-34, 2011.
- [6] Congreso Nacional de Mecatrónica, VIII. 2009. México, (2009). Control de un Robot Antropomórfico por Medio de una Interfaz con Sensores Articulares, Hernández R., Reyes F., Vargas A., et al.
- [7] Los robots aprenden el lenguaje de la cirugía. (2012) [Internet], Disponible desde: <http://www.uciencia.uma.es/Noticias/Tecnologia/Los-robots-aprenden-el-lenguaje-de-la-cirugia>>. [Acceso 20 de noviembre 2012].
- [8] Proceeding of the ACM/IEEE International Conference on Human-Robot Interaction (HRI), 6th. Suiza, (2011). Humanoid Robot Control Using Depth Camera, Suay H.
- [9] Position Location and Navigation Symposium (PLANS) IEEE/ION. USA. (2012). Study on the Use of Microsoft Kinect for Robotics Applications. El-Iaithy R., Huang J., Yeh M.
- [10] Lego MindStorm. (2012) [Internet], Disponible desde: <<http://mindstorms.lego.com>>. [Acceso 20 de noviembre 2012].
- [11] Robot Arm. (2012) [Internet], Disponible desde: <http://www.nxtprograms.com/robot_arm/steps.html>. [Acceso 20 de noviembre 2012].
- [12] Zang Z., Microsoft Kinect Sensor and Its Effect, IEEE Magazine MultiMedia, 19(2), 4 – 10, 2012.
- [13] Webb J., Ashley J., Beginning Kinect Programming with the Microsoft Kinect SDK, Apress, New York, 2012.