

# Imitación de movimientos humanos en un robot humanoide Bioloid mediante Kinect

R. Mogollán-Toral<sup>\*</sup>, O. R. Díaz-Márquez<sup>\*\*</sup>, A. Aceves-López<sup>\*\*\*</sup>

<sup>\*</sup> Instituto Tecnológico de Veracruz, México

<sup>\*\*</sup> Instituto Tecnológico de Culiacán, México

<sup>\*\*\*</sup> Instituto Tecnológico y de Estudios Superiores de Monterrey Campus Estado de México  
rolystark5@gmail.com, orlando\_rdmqz@hotmail.com, aaceves@itesm.mx

**Resumen**—La programación de movimientos en un robot humanoide resulta complicada y en ocasiones tediosa. Este artículo presenta una forma alternativa de programar movimientos en un robot humanoide mediante la imitación en tiempo real de los movimientos de un humano utilizando el sensor Kinect de Microsoft. El sistema resultante es efectivo para cualquier usuario sin importar su talla y complejidad.

**Palabras clave:** Programación de Robots, Cinemática de robots Humanoides.

## I. INTRODUCCIÓN

Todos aquellos que hayan trabajado con robots humanoides estarán de acuerdo en que una de las tareas más tediosas es programar movimientos en el robot. Básicamente se pueden clasificar en dos tipos: los movimientos que son producidos mediante alguna ecuación matemática previamente desarrollada y programada, y los movimientos que son producidos por la interpolación temporal de una serie de posturas previamente concebidas y memorizadas (Dalla et al. 2009), (Jeong et al. 2010).

Este artículo se enfoca al segundo tipo de movimientos, los cuales son empleados en humanoides para producir movimientos como levantarse del suelo, saludar, sentarse, patear una pelota, bailar o moverse suavemente mientras da un pequeño discurso. Tales acciones se ejecutan normalmente en lazo abierto, es decir, sin la necesidad de corregir en todo instante las trayectorias de brazos y piernas.

Una técnica clásica para programar dichos movimientos es descomponerlos en posturas intermedias, las cuales son obtenidas con paciencia por el programador y posteriormente probadas mediante una interpolación temporal.

Una técnica alternativa es “teaching by touching” presentada por (Dalla et al. 2009) donde se propone que un humano enseñe movimiento a un robot tocándolo directamente con la mano. Sin embargo esta técnica requiere que el robot interprete las instrucciones del humano, lo que puede llevar a ambigüedades y entonces a movimientos indeseados.

En los artículos de (Jeong et al. 2010) y (Kwon et al. 2012) se presenta una metodología para generar movimientos en un humanoide basada en demostraciones hechas por un usuario que porta una colección de sensores magnéticos. Sin embargo dichos sensores deben ser

conectados con cables y fijados en muñecas y cadera del usuario.

En el trabajo de (Yamane et al. 2010) se presenta una alternativa para programar movimientos a un robot mediante el uso de sofisticados sistemas de captura de movimiento. Sin embargo la información capturada debe adecuarse pues la cinemática y dinámica del humanoide son significativamente diferentes a las del humano.

En el artículo de (Kwon y Park 2006) se usan Modelos de Markov para encontrar una combinación de movimientos elementales que mejor representen a los movimientos demostrados por un usuario. Sin embargo, esta metodología necesita una gran cantidad de datos de entrenamiento y mucho tiempo para desarrollar un nuevo movimiento en el robot.

Otra alternativa de programar movimientos en un robot humanoide es mediante la imitación de los movimientos de un humano utilizando el sensor Kinect de Microsoft (Michieletto y Menegatti 2012). Un algoritmo muy simplificado del uso del Kinetic para mover un muñeco plano articulado puede encontrarse en (Ramos y Castro 2012). La ventaja de esta metodología es que usa un dispositivo que no requiere ser fijado con cables al usuario.

Este artículo presenta en forma detallada como lograr que un robot humanoide Bioloid imite los movimientos de un humano mediante el uso del Kinetic de Microsoft. Se analizarán las cinemáticas del humano y del robot para hacerlas comparables y se usará control PID en las articulaciones para garantizar que el robot alcanza las posturas deseadas.

La estructura del hardware es la siguiente. Una Laptop ejecuta programas hechos en lenguaje C vinculados con las librerías de Open Natural Interaction (OpenNI) para transformar las coordenadas del cuerpo humano, enviadas por el Kinetic, en ángulos articulares del robot Bioloid. Estos valores articulares serán enviados al robot mediante el protocolo de comunicación RS232. El robot Bioloid ejecuta un programa escrito en C que le permite recibir los ángulos de cada articulación, validar que no produzcan colisiones y ejecutarlos.

La sinergia de todo lo anterior dará como fruto gobernar los movimientos de un robot humanoide mediante el Kinect en tiempo real y por cualquier usuario humano; simplificando de esta manera la programación de movimientos en robots humanoides.

## II. EL KINECT DE MICROSOFT

El Kinect de Microsoft es un dispositivo popular de consolas de videojuegos que posee sensores de audio, video y de profundidad, capaces de reconocer el cuerpo humano y sus movimientos, así como el reconocimiento de voz, objetos e imágenes (TierraGamer, 2010). La figura 1 muestra sus componentes principales exteriores.

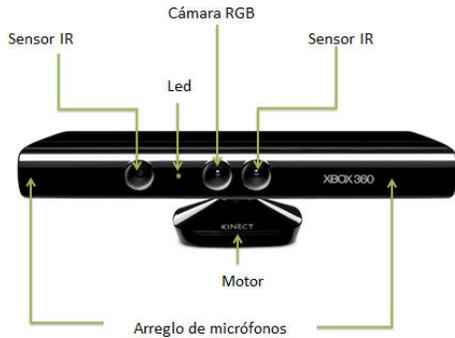


Figura 1. Partes exteriores del Kinect de Microsoft.

Recientemente Microsoft liberó en su sitio web el kit de desarrollo “Kinect for Windows SDK” que soporta lenguajes de programación en C++, C#, o Visual Basic para crear aplicaciones que necesiten de la captura de posturas y reconocimiento de voz (Kinetic, 2013).

Para lograr el rastreo del cuerpo, el Kinetic usa la información de su cámara RGB y sus sensores IR para generar una imagen RGB con información de profundidad. Gracias a esta información de profundidad y a una serie de filtros contextuales, el SKD de Kinetic puede determinar las partes principales del cuerpo (hombros, cabeza, brazos, manos y piernas) y construir una aproximación del esqueleto del humano.

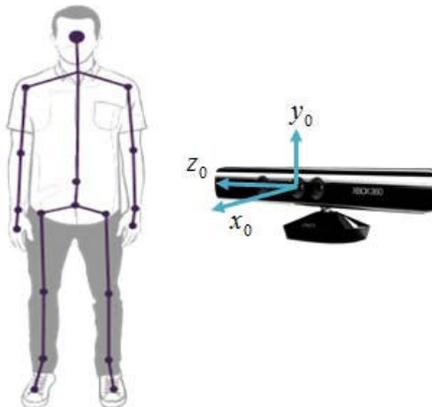


Figura 2. Sistema coordinado del Kinetic.

Es muy importante tener en cuenta que todos los puntos que entrega el Kinect están definidos en un sistema coordinado pegado a él, tal y como lo muestra la figura 2. El vector X sale al frente, el vector Y va hacia arriba y el vector Z va hacia la derecha del Kinect. A este sistema coordinado le pondremos el subíndice cero.

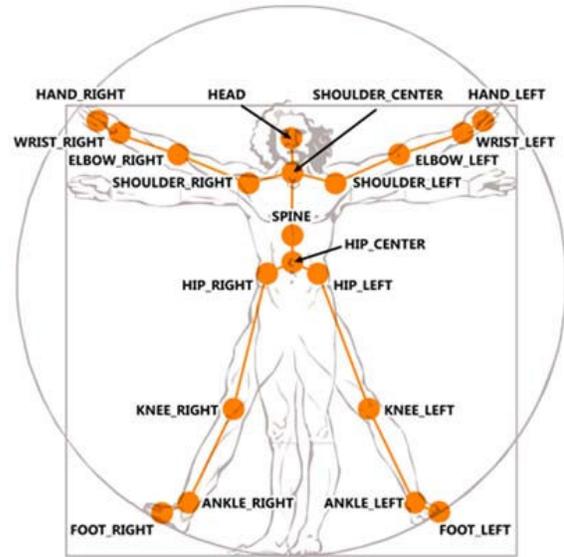


Figura 3. Puntos del cuerpo que son identificados por el programa Skeletal Tracking.

El SDK de Kinect es capaz de reconocer de manera clara veinte puntos del cuerpo humano y cada uno de ellos tiene un identificador particular (o nombre), según se puede ver en la figura 3. Cada punto tiene tres componentes en un espacio 3D. Por ejemplo el punto que define la ubicación de la cabeza se denota  ${}^0P_{HEAD}$  y cada una de sus componentes se denota como sigue:

$${}^0P_{HEAD} = \begin{bmatrix} {}^0x_{HEAD} \\ {}^0y_{HEAD} \\ {}^0z_{HEAD} \end{bmatrix} \quad (1)$$

Para conseguir las mejores lecturas de los puntos, es decir que no presenten demasiado ruido, se recomienda colocarse a 2.5 metros de distancia del Kinect.

## III. ROBOT HUMANOIDE BIOLOID

El humanoide Bioloïd de (Robotis, 2012) es un robot de bajo costo con 18 grados de libertad (3 en cada brazo y 6 en cada pierna) con un módulo de control central CM-5 capaz de gestionar los actuadores Dynamixel Ax-12 y el sensor infrarrojo Ax-S1. Dicho robot resulta muy conveniente para emular la estructura del cuerpo humano lograda con el programa Skeletal Tracking del SDK de Kinetic.

La unidad central CM-5 está diseñada para almacenar y ejecutar programas que controlan los servomotores Ax-12. En su interior se encuentra un microprocesador Atmega128 a 16Mhz con 128 KB de memoria flash, programable con el compilador GCC (the GNU Compiler Collection). El CM-5 se alimenta a una tensión de trabajo de 7 a 12 V, la placa del controlador está protegida con un fusible y encapsulada en una caja de plástico resistente con espacio para una batería recargable de 9.6V (Robotis, 2012). Además cuenta con 3 leds, 6 botones pulsadores y un puerto serial RS-232.

Para comunicar la CM-5 con la computadora es necesario hacerlo mediante un convertidor USB-to-RS232. El CM-5 cuenta con un programa residente tipo downloader que permite recibir un código previamente compilado para ser

posteriormente ejecutado. Esto último se realiza con los botones de Mode y Start.

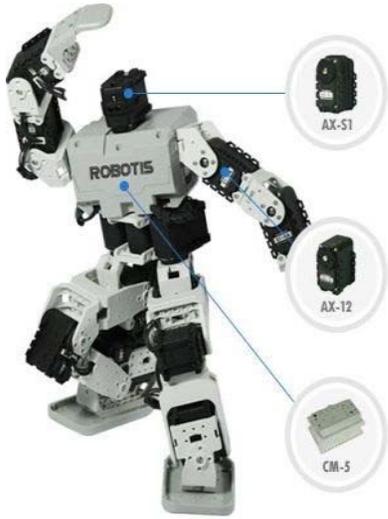


Figura 4. Las partes constitutivas del humanoide BioLoid.

Los Ax-12 son servomotores digitales los cuales se pueden comunicar con el procesador principal mediante un protocolo half duplex Asynchronous Serial (8 bit, 1 stop, No Parity) a una velocidad máxima de 1 MBPS. Internamente tienen un Atmega64 que ejecuta un algoritmo de control PID y brinda información como posición angular, velocidad angular, temperatura actual, par de carga disponible, posición actual, etc. Su rango de posiciones angulares va de 0 a 300 grados a 10 bits.

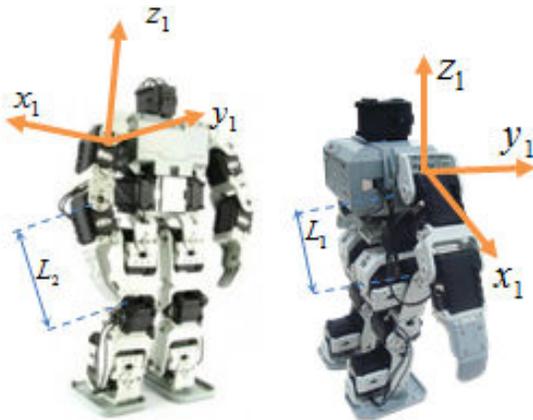


Figura 5. Sistema coordenado del brazo derecho del BioLoid.

Cada servomotor Ax-12 del BioLoid tiene un ID distinto y único, comenzando del 1 y hasta el 18. Esto permite la comunicación sin errores aunque compartan el mismo cable de comunicación.

#### IV. OPEN NATURAL INTERACTION

OpenNI es un SDK de código abierto utilizado para el desarrollo de sensores 3D, bibliotecas de middleware y aplicaciones que permiten comunicarse con los sensores de CNCA 2013, Ensenada B.C. Octubre 16-18

audio, video y sensor de profundidad del Kinect (OpenNI, 2012). Para su funcionamiento será necesario instalar sus cuatro módulos. Además, es necesario crear un nuevo proyecto de C++ por ejemplo en Microsoft Visual Studio 2010 o cualquier otro entorno de desarrollo vinculado con las librerías de OpenNI.

Con dicho software se pudo constatar que las lecturas de los 20 puntos del cuerpo corresponden consistentemente con la posición del usuario. Sin embargo, sucede que ciertos movimientos producen variaciones repentinas en las lecturas provocando posturas erróneas. Para mitigar este ruido en las lecturas, se programó un filtro simple de primer orden pasabajas.

#### V. TRANSFORMACIÓN DE COORDENADAS DEL KINECT AL BIOLOID

Primero se obtiene la cinemática directa de cada uno de los brazos del BioLoid considerando que su sistema coordenado está definido en la base del hombro. Dicha cinemática será asimilada con la del ser humano frente del Kinetic. En la figura 5 solo se muestra el sistema coordenado del brazo derecho del robot y al cual se le pondrá el subíndice uno. El vector X va hacia la derecha, el vector Y va hacia el frente y el vector Z va hacia arriba del BioLoid. Las distancias  $L_1$  y  $L_2$  del BioLoid son respectivamente 7cm y 10cm.

Utilizando la metodología de Denavit–Hartenberg se pueden establecer que la ubicación del hombro y muñeca del brazo derecho del BioLoid medidas con respecto al sistema coordenado  $\{1\}$  son:

$${}^1P_{SHOULDER} = [0 \ 0 \ 0]^T \quad (2)$$

$${}^1P_{WRIST} = \begin{bmatrix} -s_2(L_2 + L_1c_3) - L_1c_2s_3 \\ s_1(c_2(L_2 + L_1c_3) - L_1s_2s_3) \\ c_1(L_1s_2s_3 - c_2(L_2 + L_1c_3)) \end{bmatrix} \quad (3)$$

donde  $s_i, c_i$  corresponden a  $\sin(\theta_i)$  y  $\cos(\theta_i)$  respectivamente.

Ahora se modelan los movimientos del brazo derecho del humano con los mismos tres grados de libertad que tiene el BioLoid. Se adopta el mismo sistema coordenado y se definen las distancias  $L_1$  y  $L_2$ , las cuales corresponden a los largos de brazo y antebrazo del usuario.

Para la transformación de sistemas coordenados se usa la siguiente ecuación clásica de robótica (Craig, 2006):

$${}^0P = {}^0R^1P + {}^0P_{01} \quad (4)$$

Significa que todo punto  $P$  definido en el sistema coordenado  $\{1\}$ , y representado por  ${}^1P$ , se puede expresar en el sistema coordenado  $\{0\}$ , y representado por  ${}^0P$ , si se conocen respectivamente: la matriz de rotación  ${}^0R$  del sistema coordenado  $\{1\}$  con respecto al sistema coordenado  $\{0\}$  del Kinetic y el vector  ${}^0P_{01}$  que mide la distancia entre el Kinetic y el hombro del usuario medido en el sistema coordenado del Kinetic.

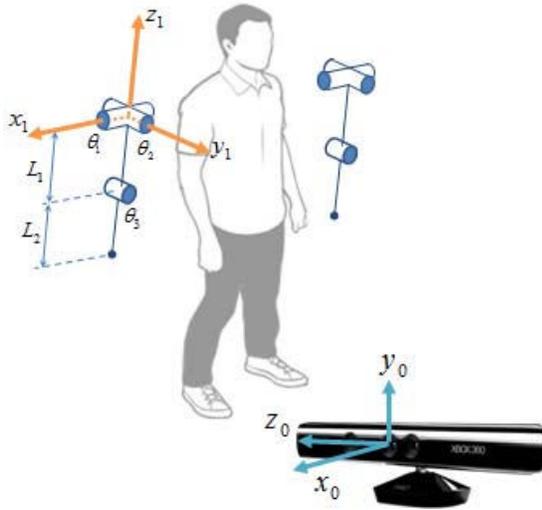


Figura 6. Sistema de coordenado del brazo derecho del humano.

Por simple inspección, y debido a que el usuario está de frente al Kinetic, se sabe que la matriz de rotación está definida como:

$${}^0R = \begin{bmatrix} {}^0x_1 & {}^0y_1 & {}^0z_1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix} \quad (5)$$

Además, el vector de traslación  ${}^0P_{01}$  se define como la ubicación del hombro derecho de la persona:

$${}^0P_{01} = {}^0P_{SHOULDER} \quad (6)$$

En efecto, despejando de la ecuación (4) se puede conocer la ubicación de la muñeca del usuario con respecto al sistema coordenado  $\{1\}$ :

$${}^1P_{WRIST} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} ({}^0P_{WRIST} - {}^0P_{SHOULDER}) \quad (7)$$

Para calcular  $L_1$  y  $L_2$  del ser humano se utilizan las siguientes ecuaciones:

$$L_1 = \left\| {}^0P_{ELBOW} - {}^0P_{SHOULDER} \right\|_2 \quad (8)$$

$$= \sqrt{({}^0x_{ELBOW} - {}^0x_{SHOULDER})^2 + ({}^0y_{ELBOW} - {}^0y_{SHOULDER})^2 + ({}^0z_{ELBOW} - {}^0z_{SHOULDER})^2}$$

$$L_2 = \left\| {}^0P_{WRIST} - {}^0P_{ELBOW} \right\|_2 \quad (9)$$

Después de tener las coordenadas de la muñeca derecha del usuario con respecto al sistema coordenado  $\{1\}$ , se deberá modelar su postura utilizando la cinemática inversa del Bioloid, y así obtener seis ángulos (tres por cada brazo).

## VI. SOLUCIÓN DE LA CINEMÁTICA INVERSA DEL BIOLOID

Igualando las ecuaciones (3) y (7) y utilizando métodos algebraicos para la solución cerrada de la cinemática inversa (Craig, 2006) se obtiene:

$$c_3 = \frac{{}^1x_{WRIST}^2 + {}^1y_{WRIST}^2 + {}^1z_{WRIST}^2 - L_1^2 - L_2^2}{2L_1L_2} \quad (10)$$

$$\theta_3 = \text{atan2}(\sqrt{1 - c_3^2}, c_3) \quad (11)$$

$$\theta_1 = \text{atan2}({}^1y_{WRIST}, -{}^1z_{WRIST}) \quad (12)$$

$$\theta_2 = \text{atan2}\left(-{}^1x_{WRIST}, \sqrt{{}^1y_{WRIST}^2 + {}^1z_{WRIST}^2}\right) - \text{atan2}(L_2s_3, L_1 + L_2c_3) \quad (13)$$

De esta manera las ecuaciones (7)-(13) constituyen la solución a la cinemática inversa del brazo derecho del Bioloid tomando como base las coordenadas enviadas por el Kinetic del brazo derecho del usuario. Nótese que los ángulos para el brazo izquierdo del Bioloid se calculan de forma equivalente.

## VII. TRANSMISIÓN DE ÁNGULOS AL CM-5

Una vez que la Laptop ha calculado los seis ángulos (tres por brazo), se envían por medio del protocolo RS232 al CM-5, el cual convierte el valor articular (recibido en grados) en el valor actuator correspondiente para cada servomotor. Esto se logra con la siguiente fórmula:

$$q_i = \max\left(0, \min\left(\frac{1024}{300}\theta_i + 512, 1024\right)\right) \quad (14)$$

Los ID's de los servomotores están determinados de la siguiente manera. Los  $\{q_1, q_2, q_3\}$  del brazo derecho tienen respectivamente los ID's  $\{1, 3, 5\}$ , mientras que los  $\{q_4, q_5, q_6\}$  del brazo izquierdo tienen respectivamente los ID's  $\{2, 4, 6\}$ .

Adicionalmente se incluyeron condiciones de validación en los valores articulares para garantizar que los eslabones de los brazos del robot no colisionan.

El programa en el CM-5 es un ciclo infinito que lee el puerto serial, convierte los valores recibidos a valores de actuator y son enviados a los seis Ax-12 correspondientes. Dicho programa se codificó en C y compiló para el CM-5 mediante las librerías libCm-5.h y Atmel Studio 6.

## VIII. RESULTADOS

El sistema completo se muestra en la figura 7. Comienza la captura de la postura del usuario mediante el Kinetic, los datos son procesados por una laptop, quien calcula los valores articulares para los brazos. Dichos valores son enviados al CM-5 del Bioloid quien actualiza la posición de los servomotores correspondientes para lograr que el humanoide alcance una postura igual al usuario.



Figura 7. Sistema total Kinetic-Bioloid.

El sistema total se pudo ejecutar a una tasa de actualización de 3 posturas por segundo. El cuello de botella se encuentra en el CM-5 quien tiene que leer del puerto RS232 los datos enviados por la PC y a su vez enviarlos a los servomotores correspondientes.

Se condujeron pruebas con diferentes usuarios y se pudo observar que el robot responde adecuadamente copiando la postura mostrada. En las figuras 8 y 9 se muestra una secuencia de posturas propuestas por el usuario y copiadas en tiempo real por el Bioloid.

### IX. CONCLUSIÓN

En este documento se detalló la manera de obtener las coordenadas del Kinect mediante OpenNI, así como el modelado de los brazos de un usuario para gobernar en tiempo real la posición de los brazos de un robot humanoide Bioloid.

La interface desarrollada ofrece ventajas claras para la programación de movimientos del robot porque permite al programador construir secuencias de movimiento enseñándole al robot mediante la ejecución de un usuario humano frente a un sensor Kinetic.

Esta forma sencilla para la programación de secuencias de movimiento contrasta con la forma tediosa de mover uno a uno los servomotores del robot hasta lograr cada postura.

Además, el sistema fue desarrollado para ser efectivo independientemente de la talla y complejión del usuario.

En este trabajo solo se modeló el movimiento de la parte superior del cuerpo, pero puede ser usado como base para futuros trabajos en donde se incluya el movimiento de las piernas a condición de mantener en todo momento la postura erguida del robot.

Finalmente se constató que un solo sensor Kinetic monitoreando el cuerpo de un usuario, se puede usar para controlar simultáneamente múltiples robots humanoides. El sistema desarrollado no solo tiene aplicaciones de programación o recreativas sin también didácticas por el hecho de ser un dispositivo novedoso y divertido, pues le permite al programador gobernar de forma sencilla cualquier otra configuración del robot Bioloid o cualquier otro sistema robótico.



Figura 8. Bioloid copiando postura de un usuario.



Figura 9. Bioloid copiando postura de un usuario distinto.

El sensor Kinetic resultó ser un dispositivo muy interesante, fácil de usar y programar que permite un nivel adicional de interacción con robots e inclusive de monitoreo y percepción del medio ambiente. Es por estas ventajas que vemos en la literatura cada vez más aplicaciones de robots que usan Kinetic (Kannan et al. 2012).

## X. AGRADECIMIENTOS

Agradecemos a la Academia Mexicana de Ciencias por la oportunidad de participar en el XXII Verano de la Investigación Científica 2012 y al Instituto Tecnológico y de Estudios Superiores de Monterrey Campus Estado de México por recibirnos en sus instalaciones.

## REFERENCIAS

- Craig, J. (2006). *Introducción a la Robótica*, 3era Edición, Prentice-Hall.
- Dalla, et al. (2009). A new paradigm of humanoid robot motion programming based on touch interpretation. *Elsevier Robotic and Automation Systems*. Vol 57, 846-859.
- Jeong, et al. (2010). Effective humanoid motion generation based on programming-by-demonstration method for entertainment robotics. *IEEE International Conference on Virtual Systems and Multimedia*. 289-292.
- Kajita et al. (2003). Biped walking pattern generation by using preview control of Zero-Moment Point. *IEEE international conference on robotics and automation*, 14-19.
- Kannan, et al. (2012). Kinetic based humanoid for rescue operations in disaster hit areas. *International Journal of Applied Engineering Research*. Vol 7, No 11, ISSN: 0973-4562.
- Kinetic (2013). Developer downloads of Kinetic for Windows, <http://www.microsoft.com/en-us/kinectforwindows/develop/>.
- Kwon y Park (2006). Using Hidden Markov Models to generate natural humanoid movements. *International Conference on Intelligent Robots and Systems*, 1990-1995.
- Kwon, et al. (2012). Framework for creating intuitive motion content for humanoid robots based on programming by demonstration. *Intech International Journal of Advanced Robotic Systems*. Vol. 9, No. 129, 1-10.
- Michieletto y Menegatti (2012). Human action recognition oriented to humanoid robots action reproduction. *CEUR Workshop on Popularize Artificial Intelligence*, Vol 860, 35-40.
- OpenNI (2012). The standard framework for 3D sensing. <http://www.openni.org/>.
- Okumura et al. (2003). Realtime ZMP compensation for biped walking robot using adaptive inertia force control. *IEEE RSJ International Conference on Intelligent Robots and Systems IROS Vol 1*, 335-339.
- Ramos y Castro (2012). *Arduino and Kinect Projects: Design, Build, Blow Their Minds*, ISBN: 1430241675. Apress Editor. New York.
- Robotis (2010). e-Manual v1.13.01, [http://support.robotis.com/en/techsupport\\_eng.htm](http://support.robotis.com/en/techsupport_eng.htm).
- TierraGamer (2010). Conoce con detalles técnicos cómo funciona Kinect, <http://www.tierragamer.com/index.php/conoce-como-funciona-kinect/>.
- Yamane, et al. (2010). Controlling humanoid robots with human motion data: experimental validation. *IEEE-RAS International Conference on Humanoid Robots*. 504-510.