

**ADVANCES IN THE CONTROL OF BIPEDAL PLATFORMS USING THE
SYSTEM *KINECT*****AVANCES EN EL CONTROL DE PLATAFORMAS BÍPEDAS USANDO EL
SISTEMA *KINECT***

**Ing. Diego Andrés López Duran, Ing. Jorge Enrique Hernández Amaya
PhD. César Augusto Peña Cortes**

Universidad de Pamplona.

Ciudadela Universitaria. Pamplona, Norte de Santander, Colombia.

Tel.: +(57) (7) 568 5303, Ext. 164.

E-mail: {lopez-07-11, jorgeehernandez} @hotmail.com, cesarapc@unipamplona.edu.co

Abstract: This paper presents the UPABOT- humanoid robot teleoperation and biped platform using the Kinect sensor. The user controls the movements of the robots through the Kinect sensor, acquiring the position of the main joints of your body. A simulation using OpenGL libraries was performed, which illustrate the skeleton view of the person and the 3D robot. By means of the inverse kinematic analysis are generated the reference signals control of the UPABOT humanoid robot and Biped robot.

Keywords: humanoid robot, biped robot, teleoperation, kinect, kinematics.

Resumen: En este artículo se presenta la teleoperación del robot humanoide - UPABOT y una plataforma Bípeda por medio del sensor Kinect. El usuario controla los movimientos de los robots por medio del sensor Kinect, adquiriendo la posición de las principales articulaciones de su cuerpo. Se realizó una simulación por medio de las librerías de OpenGL, las cuales ilustran la vista esqueleto de la persona y los robot 3D. Por medio del análisis cinemático inverso se crean las señales de referencia del control del robot humanoide UPABOT y el robot Bípedo.

Palabras clave: Robot humanoide, robot bípedo, teleoperación, kinect, cinemática.

1. INTRODUCCIÓN

El Los robots teleoperados son definidos por la NASA (1978) como: “*Dispositivos robóticos con brazos manipuladores y sensores y cierto grado de movilidad, controlados remotamente por un operador humano de manera directa o a través de un ordenador*” (Barrientos, 2007).

Para este caso gracias al desarrollo de nuevas tecnologías que han estado apareciendo alrededor mundo, se logró implementar uno de estos avances a una aplicación de teleoperación de un sistema robótico. El dispositivo *Kinect* desarrollado por

Microsoft que en un principio fue implementado exclusivamente para Juegos, se abrió camino en muchos campos tecnológicos, uno de esos campos es la robótica, en la cual se han estado dando diversas aplicaciones. Kinect tiene muchas características que han sido desarrolladas especialmente para la captura de datos del cuerpo humano, analizar sus movimientos y posiciones. Tomando esto como base se realizó el presente trabajo (SDK *Kinect* 2012).

Se desarrolló el control teleoperado del robot humanoide UPABOT, el cual tiene de 21 grados de libertad y de la plataforma Bípeda que tiene 12

grados de libertad, dicho control se realizo a través del dispositivo kinect, este sensor nos permite la captura del movimiento del usuario, realizando la programación pertinente se obtiene los datos de esqueleto de la persona, con los cuales el robot intentara emular los movimientos del usuario. Y con visualización de los movimientos que realiza el usuario y la simulación del robot humanoide en 3D en una interfaz gráfica.

2. DESCRIPCIÓN DEL ROBOT HUMANOIDE UPABOT Y LA PLATAFORMA BÍPEDA

2.1. Robot humanoide UPABOT

2.1.1. Parte mecánica

El material en que esta construido el robot es aluminio de 0.8 mm de grosor, consta de dos extremidades superiores cada una cuenta con 6 grados de libertad, dos extremidades inferiores con 4 grados de libertad y el pecho que es el punto de unión entre extremidades y parte importante del robot ya que se encuentran las tarjetas electrónicas de control (Hernández, 2012).



Fig. 1. Parte mecánica del robot UPABOT

2.1.2. Parte electrónica

El robot cuenta con: 21 servomotores, un microcontrolador de gama media (18F4550), una batería LIPO (7.4 v, 2200 mA) como fuente de alimentación para los servomotores, 4 baterías recargables nimh (1.2 v y 1400 ma) para el funcionamiento de las tarjetas de control, 3 tarjetas de control SERCOM para controlar los 21 servomotores, un dispositivo *Bluetooth* (RN-42) y un circuito de acondicionamiento, para recibir los datos con el *Bluetooth*, distribuir la alimentación y enviar los ángulos a cada una de las tarjetas SERCOM (Hernández, 2012).

2.2. Robot Bípedo

2.2.1. Parte mecánica

El material en que esta construido el robot son piezas plásticas, consta de dos extremidades

superiores cada una cuenta con 6 grados de libertad y la cadera que es el punto de unión entre extremidades (Robotis, 2006).



Fig. 2. Parte mecánica del robot Bípedo

2.2.2. Parte electrónica

El robot cuenta con: 12 robot servos Dynamixel, una tarjeta USB2Dynamixel y una fuente de alimentación externa de 6 - 10 V (Robotis, 2006).

2.3. Análisis cinemático

A continuación se presentara el análisis cinemático directo e inverso del robot humanoide UPABOT y del robot Bípedo.

2.3.1. Cinemática directa

La cinemática de un robot estudia el movimiento del mismo con respecto a un sistema de referencia si tener en cuenta las fuerzas que intervienen (Siciliano, 2008). Así, la cinemática se interesa por la descripción analítica del movimiento espacial del robot, como una función de tiempo, y en particular de la relación entre la posición y la orientación del extremo final del robot, con los valores que toman sus coordenadas articulares.

2.3.1.1. Cinemática directa del brazo

En la figura 3(a) la simulación del análisis cinemático directo de la extremidad inferior izquierda en el software Matlab y en la figura 3(b) la simulación del análisis cinemático directo de la extremidad inferior izquierda en el software *SolidEdge*. En la tabla 1 se muestran los parámetros DH obtenidos con el análisis cinemático directo del brazo (Hernández, 2012).

Tabla. 1. Parámetros DH cinemática directa.

q	d (mm)	a (mm)	A
Q1	25	-12.5	$\pi/2$
Q2+ $\pi/2$	5	0	$\pi/2$
Q3	125.7	0	$-\pi/2$
Q4	-3.5	37.2	0

Por medio de la tabla 1 se calculan todas las matrices de transformación homogéneas que relacionan el extremo del brazo con el sistema de referencia del suelo.

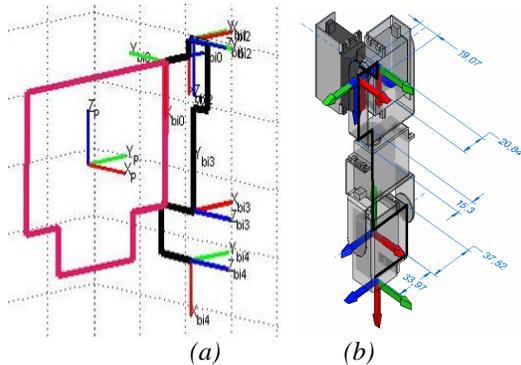


Fig. 3. Cinemática del brazo.

2.3.1.2. Cinemática directa de la pierna

En la figura 5(a) la simulación del análisis cinemático directo de la extremidad superior izquierda en el software Matlab y en la figura 5(b) la simulación del análisis cinemático directo de la extremidad superior izquierda en el software SolidEdge. En la Tabla. 2 se muestran los parámetros DH obtenidos con el análisis cinemático directo de la pierna (Hernández, 2012).

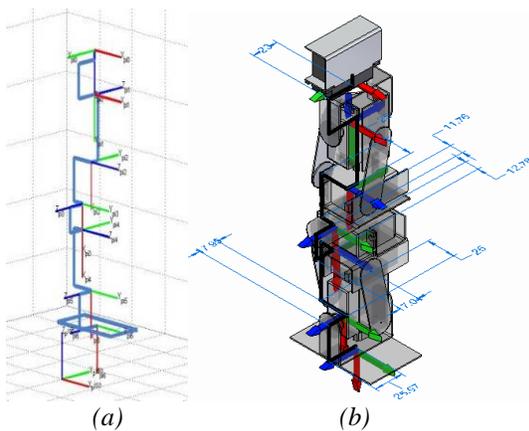


Fig. 4. Cinemática de la pierna.

Tabla. 2. Parametros DH cinemática directa de la pierna.

q	d (mm)	a (mm)	A
Q1	37.9	0	$\pi/2$
Q2+ $\pi/2$	5	-56.6	$\pi/2$
Q3	-13	37.6	$\pi/2$
Q4	0	21.9	$-\pi/2$
Q5	13.2	50	$\pi/2$
Q6	-8.9	30.9	0

2.3.2. Cinemática inversa

El problema cinemático inverso consiste en hallar los valores de las articulaciones para que el efector final del robot se posicione en unas coordenadas (x, y, z) y orientación en un punto dado en el espacio de trabajo (Siciliano, 2008).

2.3.2.1. Pierna

La cinemática inversa de las piernas se implementó en el desarrollo de los algoritmos de marcha. Para el caso específico de marcha hacia adelante, se tuvo en cuenta el tipo de movimiento para que el robot avance hacia adelante y mantenga el equilibrio, usando el centro de gravedad, dando como parámetros, X como tamaño del paso, Y es parte de la inclinación lateral y Z como altura. El análisis se basó en la utilización del centro de gravedad como referencia para que el robot avanzara y se inclinara de manera lateral.

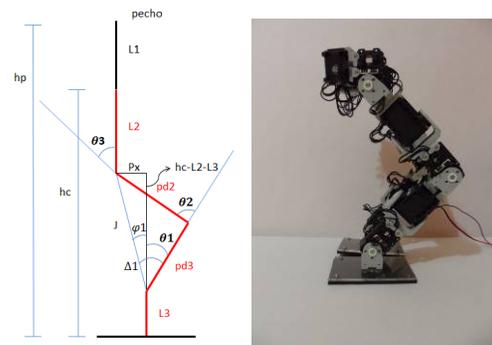


Fig. 5. Cinemática inversa de la pierna para el avance.

Teniendo en cuenta los parámetros datos de altura y tamaño del paso (p_y , p_x) del pie se puede encontrar los ángulos mediante las siguientes ecuaciones:

$$H_p = p_y \quad (1)$$

$$H_c = H_p - L_1 \quad (2)$$

$$J = \sqrt{p_x^2 + (H_c - L_2 - L_3)^2} \quad (3)$$

Por lo cual q_2 se puede calcular por medio de (4) y (5).

$$\cos q_2 = \frac{J^2 - pd3^2 - pd2^2}{2 \cdot pd2 \cdot pd3} \quad (4)$$

$$\sin q_2 = \sqrt{1 - \cos^2 q_2} \quad (5)$$

$$q_2 = \arctan 2(\sin q_2, \cos q_2) \quad (6)$$

Por lo tanto q_1 , se puede definir como la resta de (7) y (8).

$$\Delta_1 = (pd2 \cdot \sin q_2, pd2 + pd3 \cdot \cos q_2) \quad (7)$$

$$j_1 = \arctan 2(x, (H_c - L_2 - L_3)) \quad (8)$$

$$q_1 = \Delta_1 - j_1 \quad (9)$$

De esta forma q_1 queda definido como:

$$q_3 = -q_1 - q_2 \quad (10)$$

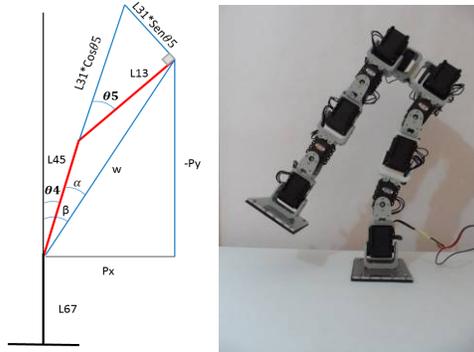


Fig. 6. Cinemática inversa de la pierna para la inclinación lateral.

Dónde:

$$W = \sqrt{px^2 + (Py - L_{67})^2} \quad (11)$$

$$W = L_{45}^2 + L_{13}^2 + 2 \cdot L_{31} \cdot L_{45} \cdot \cos(180 - q_5) \quad (12)$$

$$Py^2 + Px^2 = L_{45}^2 + L_{13}^2 + 2 \cdot L_{31} \cdot L_{45} \cdot \cos(q_5) \quad (13)$$

Teniendo los valores de (13), q_5 se puede calcular según (14) y (15).

$$\cos q_5 = \frac{Py^2 + Px^2 - L_{45}^2 - L_{13}^2}{2 \cdot L_{31} \cdot L_{45}} \quad (14)$$

$$\sin q_5 = \sqrt{1 - \cos^2(q_5)} \quad (15)$$

$$q_5 = \arctan 2(\sin q_5, \cos q_5) \quad (16)$$

Con (15), el valor de q_4 se calcula según (17) y (18).

$$j_1 = \arctan 2\left(\frac{L_{13} \cdot \sin q_5}{L_{45} + L_{13} \cdot \cos q_5}\right) \quad (17)$$

$$b = \arctan 2\left(\frac{-Py}{Px}\right) \quad (18)$$

$$q_4 = b - a \quad (19)$$

2.3.2.2 Brazo

Para encontrar los ángulos de la extremidad inferior, se adquieren los datos de la posición de la mano, referenciada al hombro como sistema base, de esta manera se realiza el análisis cinemático inverso implementado el método geométrico.

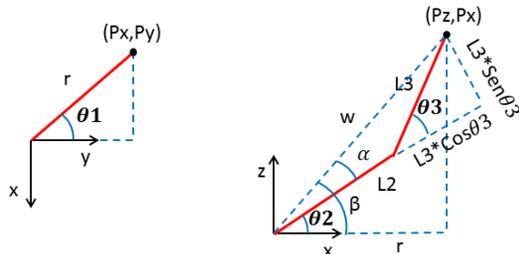


Fig. 7. Cinemática inversa del brazo.

Por lo cual q_1 , se define como:

$$q_1 = \arctan 2\left(\frac{-Py}{Px}\right) \quad (20)$$

El valor de q_2 , se calcula a partir de (21-23):

$$r = \sqrt{px^2 + py^2} \quad (21)$$

$$j_1 = \arctan 2\left(\frac{L_3 \cdot \sin q_3}{L_2 + L_3 \cdot \cos q_3}\right) \quad (22)$$

$$b = \arctan 2\left(\frac{-Pz}{r}\right) \quad (23)$$

$$q_2 = b - a \quad (24)$$

Por lo cual es necesario calcular en primera instancia q_3 según (25-27):

$$h = \sqrt{pz^2 + px^2} \quad (25)$$

$$\cos q_3 = \frac{h^2 - L_2^2 - L_3^2}{-2 \cdot L_2 \cdot L_3} \quad (26)$$

$$\sin q_3 = \sqrt{1 - \cos^2 q_3} \quad (27)$$

$$q_3 = \arctan 2(\sin q_3, \cos q_3) \quad (28)$$

2.3.2. Centro de masa

Utilizando el centro de masa generado con una posición específica se logró el análisis de estabilidad del robot humanoide UPABOT, con esto se buscó mejorar los movimientos realizados por el robot en cada una de sus rutinas.

Para esto fue necesario desarrollar un algoritmo para establecer el centro de masa del robot y su proyección en el suelo para evitar que se alejara de los puntos de apoyo, dependiendo del ángulo de sus articulaciones (Arbulú, 2008).

En la Fig. 8 los puntos azules ilustran los centros de masa de eslabones y los puntos rojos los centros de masa por extremidad. El punto verde indica la proyección centro de masa total del robot sobre el suelo.

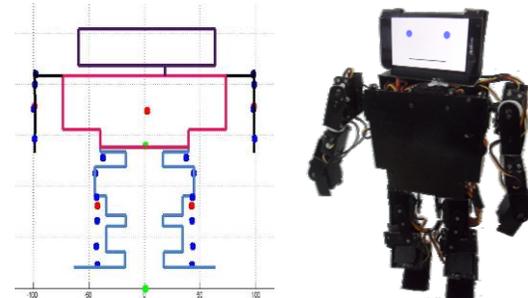


Fig. 8. Centros de masa del robot

3. TELEOPERACIÓN DEL ROBOT

A continuación se muestran la teleoperación de robot humanoide y del robot bípedo, los protocolos de comunicación y la interfaz de usuario:

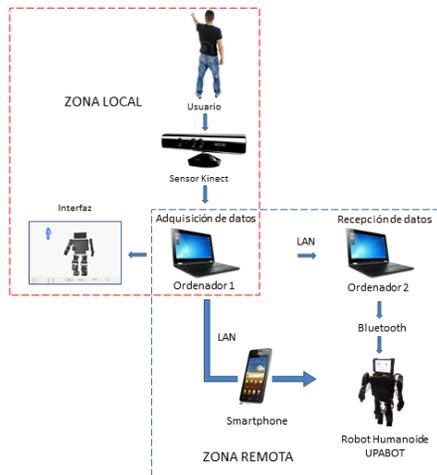


Fig. 9. Teleoperación del robot UPABOT con Kinect.

En la estación de trabajo local se utilizó el sensor *Kinect* el cual captura los movimientos realizados por el usuario, un ordenador en donde se reciben y se simulan en la interfaz de usuario los datos obtenidos con el sensor (ver Fig. 9 y Fig. 10).

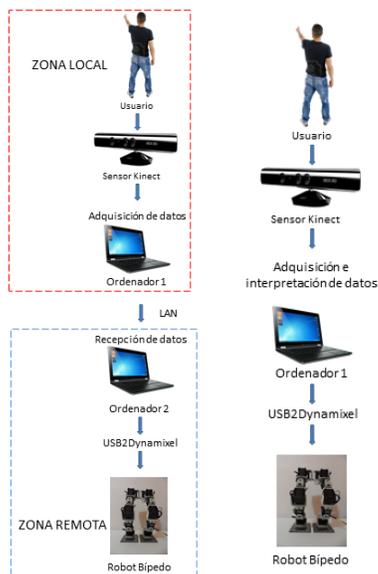


Fig. 10. Teleoperación del robot Bípedo con Kinect.

El software utilizado para realizar la adquisición y visualización de los datos, en este caso es Visual Studio 2010, la interfaz de usuario permite monitorear los puntos de esqueleto que el sensor

captura y una simulación en 3D del robot humanoide con sus respectivos movimientos.

La estación remota se puede comunicar de dos maneras las cuales son: comunicación UDP en donde la recepción de los datos la realiza un ordenador o *Smartphone* y comunicación vía *Bluetooth* en donde la recepción de datos la realiza un módulo *Bluetooth* que tiene el robot, independiente de la comunicación utilizada, es el enlace entre la estación local y la estación remota que es la encargada de realizar el control del robot.

3.1. Algoritmo desarrollado

El software Visual Studio 2010 fue utilizado para realizar la interfaz de usuario, en el entorno MFC (*Microsoft Foundation Classes*) se desarrollo la programación en el lenguaje C++, primero la adquisición de datos del esqueleto que se obtiene con el uso del sensor *Kinect*, con los datos obtenidos y analizados, se realiza el control cinemático de la extremidades y las simulaciones de la vista esqueleto del usuario y modelo 3D del robot UPABOT.



Fig. 11. Algoritmo del robot UPABOT.

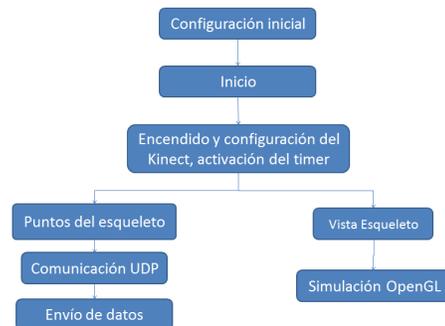


Fig. 12. Algoritmo de la plataforma Bípedo.

Las simulaciones se realizaron para monitorear los movimientos del robot en 3D y del operador con la vista esqueleto (Hearn, 2006).

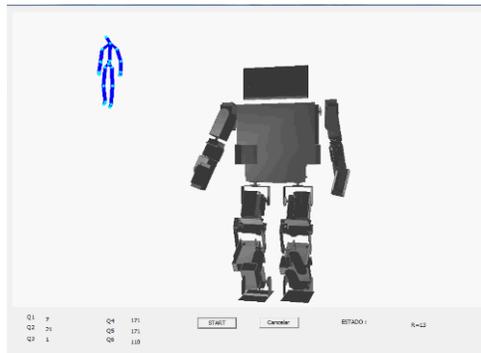


Fig. 13. Interfaz de usuario.

3.2. Algoritmo desarrollado en Matlab

El software Matlab se utilizó para recibir los datos de obtenidos con Visual Studio 2010 y permitió desarrollar los algoritmos de marcha.

3.2.1. Algoritmos de marcha

Las rutinas pre-programadas o algoritmos de marcha, se realizaron utilizando la cinemática inversa de cada una de las extremidades superiores, de acuerdo a cada movimiento requerido por la rutina. Alguno de los movimientos realizados con la cinemática son la inclinación lateral, apoyo en un solo pie, avance hacia del pie hacia adelante o hacia atrás, flexión de la extremidad. Los algoritmos de marcha se desarrollaron utilizando la cinemática de cada extremidad, para modificar la altura y el avance del pie (Tabla 3).

En las figuras 14 y 15 se presentan los diagramas de los algoritmos implementados para el robot UPABOT y el bípodo.

Tabla. 3. Implementación de la cinemática

	<p>Posición Cero Robot totalmente estirado.</p>
	<p>Posición inicial Se flexiona el robot modificando la altura en Y.</p>
	<p>Posición de avance Se flexiona el robot modificando la altura en Y y se realiza el avance del pie derecho modificando el parámetro en X.</p>



Fig. 14. Diagrama del algoritmo del Robot UPABOT



Fig. 15. Algoritmo del Robot Bípodo.

Tabla. 4. Algoritmo de marcha hacia atrás desarrollado con la plataforma bípoda.

Posiciones de la marcha atrás	Descripción
	<p>Posición inicial del robot: Ambas piernas se encuentra a la misma altura en Y y no se realiza avance en X.</p>
	<p>Posición preparación de apoyo sobre el pie derecho: La pierna derecha tiene la misma altura inicial con una inclinación lateral derecha, la pierna izquierda aumenta la altura y tiene la misma inclinación lateral de la pierna derecha.</p>
	<p>Posición de apoyo sobre el pie derecho: Aumenta la inclinación lateral de la pierna derecha, y la pierna izquierda mantiene la posición anterior.</p>

	<p>Posición de apoyo sobre el pie derecho 2: La pierna derecha mantiene la misma inclinación lateral anterior, y la pierna izquierda disminuye la altura en Y manteniendo la inclinación lateral.</p>
	<p>Apoyo sobre pie derecho y retroceso del pie izquierdo: La pierna derecha mantiene la misma altura e inclinación lateral, y la pierna izquierda mantiene la altura y se realiza un retroceso en X.</p>
	<p>Apoyo sobre pie derecho adelante y descenso del pie izquierdo atrás: Se mantiene la altura en ambas piernas, y la inclinación lateral retorna a la posición inicial.</p>
	<p>Retroceso de la cadera con Apoyo sobre pie derecho adelante y pie izquierdo atrás: Se lleva la cadera un poco mas atrás, esto se logra con un avance del pie derecho, y llevando a la pierna izquierda a la posición inicial, estos dos movimientos al tiempo.</p>
	<p>Posición preparación de apoyo sobre el pie izquierdo: La pierna izquierda tiene la misma altura inicial con una inclinación lateral izquierda, la pierna derecha aumenta la altura y tiene la misma inclinación lateral de la pierna izquierda.</p>
	<p>Posición de apoyo sobre el pie izquierdo: Aumenta la inclinación lateral de la pierna izquierda, y la pierna derecha mantiene la posición anterior.</p>
	<p>Posición de apoyo sobre el pie izquierdo 2: La pierna izquierda mantiene la misma inclinación lateral anterior, y la pierna derecha mantiene la altura en Y, realiza un retroceso en X manteniendo la inclinación lateral.</p>
	<p>Posición Final: Retorna a la posición inicial tomada por el robot.</p>

En la tabla 4 se presentan los pasos que sigue el algoritmo de marcha desarrollado para la plataforma bípeda.

Con la teleoperación del robot humanoide UPABOT se puede realizar acciones como caminar hacia adelante o atrás, flexionar las piernas, moverse a los lados y mover los brazos.

En la Fig. 16 se puede mirar el movimiento del usuario, levantando el brazo izquierdo, la interfaz de usuario con la simulación del robot y la plataforma humanoide emulando los movimientos del operador.



Fig. 16. Teleoperación del Robot humanoide UPABOT con kinect

Con la teleoperación del robot Bípedo se puede realizar acciones como caminar hacia adelante o atrás, flexionar las piernas, moverse a los lados y levantar un pie.

En la Fig. 17 se puede observar el movimiento del usuario, levantando el pie izquierdo, la interfaz de usuario con la simulación de los puntos que captura el sensor Kinect y la plataforma Bípeda emulando los movimientos del operador.

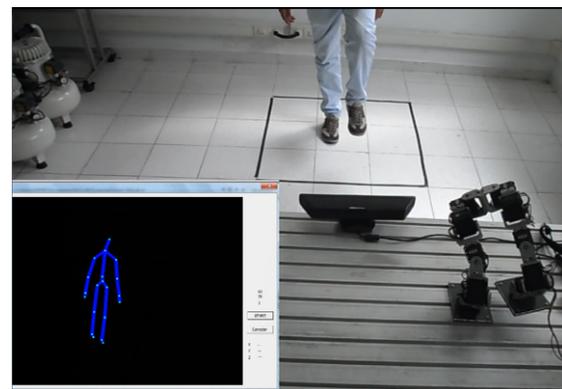


Fig. 17. Teleoperación del Robot Bípedo con Kinect

5. CONCLUSIONES

Se logró establecer la teleoperación del robot desde la estación local, en la cual se realiza la captura de datos de posición del operador utilizando *Kinect*, con la estación remota, en la cual se controla el robot humanoide UPABOT o el robot Bípido.

Se establecieron algoritmos para la captura de puntos correspondientes al cuerpo del usuario en el espacio con el sensor *Kinect* con el fin de generar las referencias en la teleoperación de los robots.

El algoritmo de programación que tiene en cuenta el centro de masa de la plataforma humanoide logra una buena estabilidad en los movimientos del robot UPABOT.

La integración de los algoritmos de marcha, utilizando la cinemática, la captura de los puntos del cuerpo del usuario, la vista esqueleto permitieron la generación correcta de comandos de alto nivel en los robots, tales como: caminar, desplazarse hacia el lado izquierdo, derecho y agacharse.

La simulación tridimensional del robot humanoide UPABOT implementada utilizando las librerías de *OpenGL*, y desarrolladas desde un *CAD* generan una percepción bastante intuitiva del robot el cual puede estar ubicado en una zona remota.

La interfaz de usuario desarrollada, permite la supervisión y control, ya que en esta se observa algunos datos como ángulos, puntos de interés, se visualiza los movimientos del operador y la simulación del robot en 3D.

El desarrollo de este dos prototipos dentro del marco de un semillero de investigación, incentivo a los estudiantes en el aprendizaje de varios conceptos fundamentales en robótica.

REFERENCIAS

- Barrientos, A., Peñin L., Balaguer C., Aracil R. (2007). *Fundamentos de Robótica*, 2ª edición, McGraw-Hill, Madrid, España, p. 13.
- Hernández A, J (2012). *Robot humanoide UPABOT teleoperado*, Jorge Enrique, Universidad de Pamplona, Pamplona, Colombia.
- Vega R, E D, (2012). *Teleoperación de un robot antropomórfico por medio de kinect*. Universidad de Pamplona, Pamplona, Colombia.
- Siciliano. B, Khatib. O. (2008) “*Springer HANBOOK OF ROBOTICS*” Springer – Velag, Berlin.
- Fernández J. A, (2002). *Manipulador virtual teleoperado*, Instituto Militar de Ingeniería, Río de Janeiro.
- Ollero. A. (2001) *Robótica: Manipuladores y robots móviles*. Marcombo.
- Hearn D., Baker M., (2006). *Graficos por computadora con OpenGL*, Indiana University – Purdue University, edición 3, Prentice Hall.
- Christensen J,(2007). *Development, Modeling and Control a humanoid robot*, Aalborg University.
- Souères, P y Laumond, J (2010). *Modeling of human movement for the generation of humanoid robot motion*, institut national polytechnique de Toulouse.
- Arbulú S, M R. (2008). *Stable locomotion of humanoid robots based on mass concentrated model*, Universidad Carlos III de Madrid.

SITIOS WEB

- Microsoft SDK Kinect for Developers (2012), Kinect for Windows:*
<http://www.microsoft.com/enus/kinectforwindow>.