

# CONTROL DE UN ROBOT HUMANOIDE UTILIZANDO MOTION RETARGETING

## CONTROL OF A HUMANOID ROBOT USING MOTION RETARGETING

Juan Felipe Medina Lee<sup>1</sup>, Luis Miguel Capacho Valbuena<sup>2</sup>

---

<sup>1</sup> Universidad del Quindío, Armenia, Quindío. E-mail: jfmedina@uniquindio.edu.co

<sup>2</sup> Universidad del Quindío, Armenia, Quindío. E-mail: lmcapacho@uniquindio.edu.co

Recibido: 3 Noviembre

Aceptado: 15 Diciembre

\*Correspondencia del autor. Universidad del Quindío, Armenia, Quindío. E-mail: jfmedina@uniquindio.edu.co

### RESUMEN

En este proyecto se hace el control de las extremidades superiores de una plataforma robótica humanoide utilizando la técnica de motion retargeting, que redirige los movimientos de un operador humano al androide. El sistema propuesto hace uso del microsoft kinect para reconocer las articulaciones del operador, posteriormente hace un cambio de coordenadas, verificación de los límites de los ángulos permitidos por la plataforma y finalmente se transmiten los movimientos al robot para que 'imite' los movimientos del operador, actualmente el sistema reproduce dos grados de libertad en cada hombro, uno en cada codo y uno en la cintura. Con este proyecto se pueden realizar rutinas de movimientos para la plataforma sin necesidad de marcas especiales y/o cables sobre el operario, y se evitan procedimientos invasivos

**Palabras claves:** Control, kinect, motion retargeting, plataforma robótica, robot humanoide.

### ABSTRACT

This project implements the control of upper extremities in a humanoid robotic platform using a motion retargeting technique. The system uses a Microsoft Kinect to recognize the operator's joints; then it performs a coordinate's transformation, validates the limit angles of the robotic platform and finally it transmits the movements to humanoid platform. The robot has two degrees of freedom in the shoulder, one in the elbow and another one in the waist. Using this project, several movements' routines can be played by the humanoid robot without using any special marks or cables on the operator, avoiding invasive methods to do so.

**Keywords:** kinect, motion retargeting, robotic platform, control, humanoid robot.

## Introducción

Un humanoide es un robot con forma de cuerpo humano. Actualmente existe un amplio estudio relacionado con esta área gracias al potencial para tareas diarias en el hogar, hospitales u oficina que pueden realizar (1).

El control de un robot humanoide esta enmarcado en la manipulación de ordenes preestablecidas, las cuales pueden ser enviadas por medio de interfaces físicas y eléctricas para la ejecución de un movimiento determinado. Para ello se ha usado interfaces de comunicación alámbrica e inalámbricas.

Este tipo de control dificulta la fluidez en el movimiento adecuado del humanoide (2). Un control mas preciso usado en la actualidad consiste en realizar la manipulación del humanoide usando como referencia los movimientos de las articulaciones de un ser humano. Para la obtención de los movimientos humanos como referencia, es posible realizarlo con el uso de sensores distribuidos a lo largo de las articulaciones de la referencia. Sin embargo, se ha optado por hacer uso de captura de movimiento usando procesamiento de imágenes para la obtención de puntos de interés.

En este trabajo se muestra el control de un robot humanoide usando el microsoft kinect que permite la captura de las articulaciones humanas para luego enviar los movimientos al robot humanoide que son imitados manipulando las mismas articulaciones en el robot y permitiendo una posición correspondiente del efector final con respecto a la referencia humana; este concepto es conocido como motion retargeting.

## Componentes del sistema

### Cámara y captura de datos

La captura de datos se hace utilizando la plataforma kinect de Microsoft, se eligió debido a su capacidad de capturar imágenes con profundidad y al soporte de software que tiene, ya que con el SDK proporcionado por Microsoft (3) se puede extraer fácilmente información de color, profundidad y ubicaciones de las articulaciones, especialmente las de la parte superior del cuerpo, en donde se pueden capturar 11 grados de libertad (GDL) (4) como se muestra en la Figura. 1

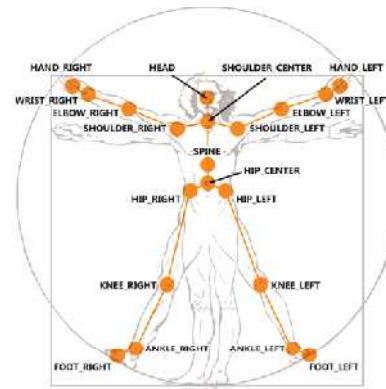


Figura 1: Articulaciones capturadas por el kinect

### Plataforma robótica

La plataforma robótica utilizada en este proyecto se muestra en la Figura. 2 Su nombre es ANDI (androide interactivo), desarrollado en la ciudad de Armenia por la empresa Roboti-k. Es un robot humanoide que tiene cinco GDL en cada brazo, dos GDL en la cabeza y uno en la cintura. El robot tiene la capacidad de desplazarse utilizando un modelo con llantas diferencial. Los servomotores utilizados en el robot son Dinamixel RX-24F para los hombros y codos y AX-12 en el resto de las articulaciones.

### Software

El software utilizado para el desarrollo del proyecto fue Visual Studio 2010, por su compatibilidad con el SDK de Microsoft para el sensor Kinect. El proyecto fue desarrollado en C#. El protocolo de comunicación serial RS232 fue usado para la comunicación con el robot, usando un puerto virtual a 9600 baudios, con 1 bit de parada y sin bits de paridad.



Figura 2: ANDI, androide interactivo

## Problema de imitación

Hay que tener ciertas consideraciones en cuenta cuando se va a trasladar movimiento desde un operario; esto debido a las diferencias físicas entre el brazo humano y el brazo de la plataforma robótica, éstas pueden ser de velocidad, amplitud o simplemente cantidad en grados de libertad. Por ejemplo, el hombro humano es una de las articulaciones más complejas del cuerpo, ya que tiene tres grados de libertad, dos de los cuales tienen un rango de, estas características son muy difíciles de igualar para un androide, razón por la cual se deben imponer ciertas restricciones. A continuación se hace un breve análisis sobre las articulaciones de un brazo humano y del brazo de ANDI.

## Articulaciones humanas

El brazo humano tiene 6 GDL. Los tres primeros se encuentran en la articulación del hombro, uno en el codo y dos en la muñeca. Los movimientos de pitch y de roll tienen una amplitud de  $90^\circ$ , mientras que el de yaw es de  $180^\circ$  para el hombro. El GDL del codo puede tener inclinaciones de casi  $180^\circ$ . Para la muñeca se tiene en el GDL yaw y  $90^\circ$  para el pitch. Una descripción gráfica de los grados de libertad de un brazo humano se muestran en la Figura. 3

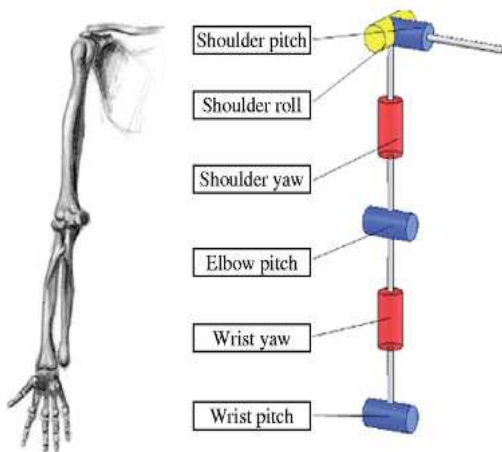


Figura 3: Grados de libertad del brazo humano [5]

## Articulaciones del robot

La plataforma que se utilizó en este proyecto tiene en total 13 articulaciones, 5 de ellas en cada brazo, 2 en la cabeza y 1 en la cintura como se muestra en la Fig 4. Dependiendo de la articulación se utilizan diferentes tipos de servomotores, para los hombros y codos se utilizan Dynamixel RX-24F y para las demás articulaciones Dynamixel AX-12.



Figura 4: Grados de libertad de ANDI

En el hombro de ANDI se tienen 2 GDL (pitch y roll), en el codo tiene 1 GDL y en la muñeca 2 GDL; de modo que no se tiene una diferencia muy grande en ese sentido con un brazo humano, ya que la única diferencia es el yaw del hombro, donde se tiene una diferencia más notable es en las amplitudes de los movimientos, ya que en el robot, se tiene un rango de  $90^\circ$  en los hombros y de  $180^\circ$  en el codo, frente a los casi  $180^\circ$  de rango en estos mismos movimientos del cuerpo humano.

## Cálculo de los ángulos

Para que el robot mueva una articulación, se debe transmitir el ID del servo que se quiere mover, el ángulo destino y la velocidad del movimiento. Por esta razón lo primero que se debe hacer es calcular los ángulos de cada uno de los movimientos que se van a enviar. Con la lectura del Kinect se tienen las coordenadas X,Y,Z de cada una de las articulaciones, y a partir de éstas se pueden calcular los ángulos deseados utilizando las distancias entre articulaciones y fórmulas trigonométricas. El ángulo del codo se calcula utilizando el producto punto de dos vectores, finalmente se obtiene (1)

$$\theta_1 = \pi - \cos^{-1} \left( \frac{l_1^2 + l_2^2 - \|p\|^2}{2l_1 l_2} \right) \quad (1)$$

Donde es la distancia euclidiana entre el hombro y el codo, entre el codo y la muñeca; y es la distancia entre

el hombro y la muñeca. Al ser distancias euclidianas entre dos puntos en tres dimensiones se calculan utilizando (2)

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2} \quad (2)$$

En la Formula 2. se observan las distancias necesarias para el cálculo del ángulo  $i_4$

El cálculo para los ángulos de los hombros es mas sencillo, ya que sólo se necesitan dos articulaciones (hombro y codo) que se obtienen utilizando fórmulas trigonométricas básicas, dependiendo del GDL que se quiera calcular, se utilizan ciertas coordenadas de las articulaciones, los ángulos para los dos GDL del hombro son calculados en (3) y (4)

$$i_1 = \cos^{-1} \left( \frac{c_y}{\sqrt{c_x^2 + c_y^2}} \right) \quad (3)$$

$$i_2 = \cos^{-1} \left( \frac{c_z}{\sqrt{c_x^2 + c_y^2}} \right) \quad (4)$$

Donde  $c_x$ ,  $c_y$  y  $c_z$  son las diferencias entre las coordenadas del codo y las del hombro. En la Figura. 5 se observa el esquema de las articulaciones y ángulos del hombro y el codo en el brazo humano.

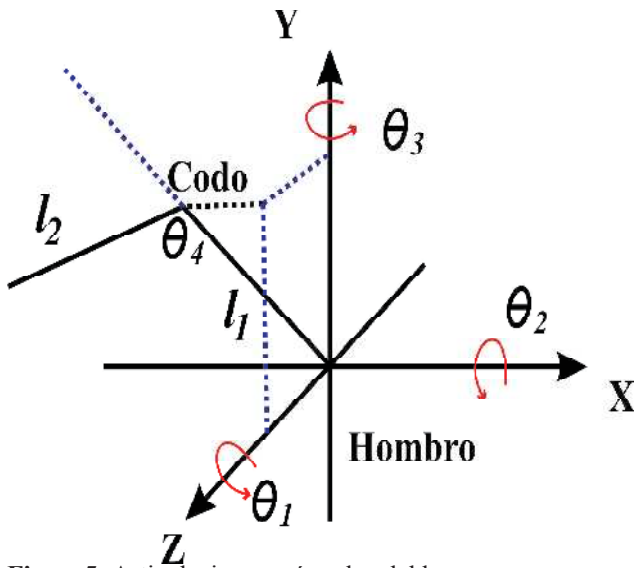


Figura5: Articulaciones y ángulos del brazo

Por restricciones de ANDI, el ángulo  $i_3$  no es calculado, ya que el robot no cuenta con este movimiento.

**Mapeo de los ángulos**

Como se mencionó anteriormente, la principal diferencia entre el robot y el cuerpo humano, son los rangos de movimientos en los GDL, que en el robot son mucho más pequeños, razón por la cual se deben poner ciertas reglas para evitar daños en el robot en-

viándole ángulos que no puede lograr o que realice movimientos no deseados.

La primera de estas reglas es la limitación de los ángulos, para cada uno de los GDL en hombros y codos del robot se cumple la regla presentada en (5)

$$i_r = \begin{cases} 0, & \text{si } i_h < 0, \\ i_h, & \text{si } 0 < i_h < i_{\max} \\ i_{\max}, & \text{si } i_h > i_{\max} \end{cases} \quad (5)$$

Para el caso de los hombros,  $i_{\max} = 70^0$  y para los codos,  $i_{\max} = 60^0$ .

Otra regla que se utiliza para el mapeo de ángulos en los hombros minimiza considerablemente movimientos no deseados en los ángulos  $i_1$  y  $i_2$ . Esta regla se utiliza para disminuir el efecto de las oclusiones en ciertos movimientos de la persona ya que pueden ocasionar la obtención de ángulos errados por parte del kinect. Esta regla es mostrada en (6) y (7).

$$i_1 = 0.6c_x \cdot 1.10 \quad (6)$$

$$i_2 = 0.6c_z \cdot 1.10 \quad (7)$$

**Adaptación de los datos**

La tasa de captura para esta aplicación con el Kinect es de 15fps aproximadamente, en cada frame capturado se calculan los ángulos mencionados anteriormente, pero no todos son enviados al robot porque se pierde fluidez en los movimientos, las coordenadas de las articulaciones no siempre son acertadas y pueden haber ángulos inexactos que ocasionen movimientos bruscos en la plataforma o simplemente porque se puede estar reconociendo el esqueleto equivocado. Por estas razones se implementaron diferentes técnicas de optimización que mejoran el rendimiento del motion retargeting.

**Filtro mediana**

Un filtro mediana de longitud 19 es aplicado a cada uno de los 6 ángulos obtenidos, de tal forma que se descartan algunos ángulos productos del ruido del Kinect que ocasionarían movimientos bruscos en la plataforma. De esta forma se reduce notoriamente el ruido a cambio de un pequeño lag en la respuesta del sistema.

## Reconocimiento de un solo esqueleto

El Kinect es capaz de reconocer más de un esqueleto simultáneamente, si no se tiene una regla clara de cuál de ellos manejará el robot, se podrían presentar movimientos inadecuados, oclusiones y acciones involuntarias que podrían causar problemas en la plataforma. La forma de elegir una sola persona es dibujando una cara en la imagen, la persona que ubique su cara sobre la cara de la imagen obtiene el control de la plataforma hasta que salga de la imagen, ninguna otra persona que aparezca en escena tendrá ningún efecto sobre el robot.

## Suavizado de movimientos

Si se enviaran todas las posiciones obtenidas y suavizadas con el filtro mediana, para las trayectorias largas se obtendrían una serie de "micro-movimientos" llevarían la articulación paso a paso hasta el ángulo final, lo que tendría como resultado una fluidez muy baja en los movimientos amplios. Para evitar el efecto de los movimientos paso a paso se implementó una técnica que considera la diferencia entre el ángulo actual y el ángulo anterior para enviar los movimientos al robot, el algoritmo es mostrado en la Figura. 6

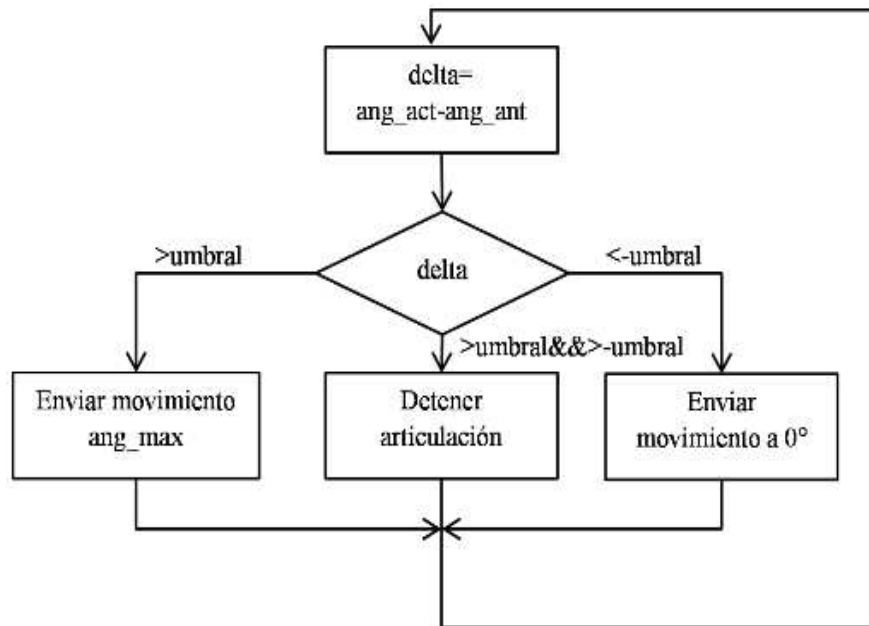


Figura 6: Diagrama de flujo para el suavizado de los movimientos

De esta forma, en cada iteración se compara el ángulo actual con el anterior y la acción enviada al robot solo es actualizada si cambia el estado en la variable delta, de modo que si, por ejemplo, el hombro del operario continúa subiendo entre varios frames, al robot solo se envía una vez que vaya al ángulo máximo, una vez se detenga el movimiento del operario, la variable delta será menor a un umbral y se enviará la orden de detener el movimiento en esa articulación.

## Resultados

Finalmente, después de obtener, mapear y transmitir los ángulos oportunos al robot, se logró enviar los movimientos de un operario a una plataforma robótica utilizando el Kinect como medio de captura, el sistema tiene un lag menor a 1 segundo, que para estos

fin es aceptable, ya que no se requiere una respuesta instantánea de la plataforma. En la Figura. 7 se muestra como la plataforma imita los movimientos de los brazos del operario.



Figura 7: Resultado final

## Conclusiones

Se implementó satisfactoriamente un algoritmo de motion retargeting para manipular un robot humanoide inalámbricamente utilizando el Microsoft Kinect.

Se implementaron diferentes técnicas de acondicio-

namiento de la señal para realizar movimientos más fluidos en la plataforma, como el filtro mediana y la técnica de cambio de velocidad con un ángulo fijo.

Se implementó una interfaz gráfica en lenguaje de programación C# en el IDE visual studio que permite empezar la redirección de los movimientos de manera fácil e intuitiva

---

## BIBLIOGRAFÍA

1. V. V. Nguyen and J.-H. Lee, Full-body imitation of human motions with kinect and heterogeneous kinematic structure of humanoid robot. In System Integration (SII), 2012 IEEE/SICE International Symposium on, Dec 2012, pp. 93–98.
2. N. Pollard, J. Hodgins, M. Riley, and C. Atkeson, Adapting human motion for the control of a humanoid robot. In Robotics and Automation. 2002. Proceedings. ICRA '02. IEEE International Conference on, vol. 2. 2002. pp. 1390–1397 vol.2.
3. Kinect for windows. [Internet]. Microsoft. [Consultado en 2014]. Disponible en: <http://www.microsoft.com/en-us/kinectforwindows/>
4. H.-H. Hsu, Y. Chiou, Y.-R. Chen, and T. Shih, Using kinect to develop a smart meeting room. In Network-Based Information Systems (NbiS), 2013 16th International Conference on Sept 2013. pp. 410–415.
5. A. de Estudiantes Reset. [Internet]. [Consultado en 2010]. Brazo robótico. Disponible: <http://reset.etsii.upm.es/es/projects/robotic-arm/>