

Algoritmo de captura de movimiento basado en visión por computador para la teleoperación de robots humanoides.

J.M. GARCIA

RoboticsLab, Universidad Carlos III de Madrid, jgarciah@ing.uc3m.es

S. MARTINEZ

RoboticsLab, Universidad Carlos III de Madrid, scasa@ing.uc3m.es

Resumen

Desde hace tiempo, los seres humanos estamos desarrollando robots que puedan ayudarnos a mejorar nuestra calidad de vida. Esta ayuda puede ofrecerse de distintas maneras, ya sea con la sustitución completa de la persona para eliminar completamente cualquier riesgo o con la idea de compartir distintas tareas. Desde el grupo de investigación RoboticsLab de robots humanoides en la universidad Carlos III de Madrid, estamos empezando a sentar las bases de desarrollo de un robot humanoide con la habilidad de realizar tareas de camarero. Por ello, esta investigación es una primera fase, la cual se centra en la búsqueda de un método de teleoperación que nos permita posteriormente enseñar al robot humanoide comportamientos o trayectorias relacionadas con el ofrecimiento de bebidas. Para ello, a través de la cámara Kinect y sus herramientas, hemos desarrollado un algoritmo de teleoperación basado en RPY (roll-pitch-yaw) para la obtención de los valores de actuación que a posterior se aplicaran sobre el robot humanoide TEO.

Palabras clave: Robot camarero, Visión, Teleoperación, Esqueletización, Roll Pitch Yaw, Kinect.

1. INTRODUCCIÓN

Aprovechando las cualidades que nos ofrece la visión por computador, se pretende asentar las bases para el desarrollo de una base de datos a futuro donde se pueda almacenar diferentes movimientos o trayectorias de manipulación que, a posteriori, un robot humanoide pueda ejecutar de forma autónoma y automática.

La generación de esta base de datos comienza con la implementación de una aplicación que sea capaz de capturar y almacenar los movimientos que realizar una persona. Además, este artículo se quiere enfocar a próximas investigaciones relacionadas con el robot TEO, el cual pretende realizar tareas iguales o parecidas a las de un camarero sirviendo bebidas sobre una bandeja.

Es en este punto donde la teleoperación entra en esta investigación como parte fundamental, para la generación de tareas, movimientos o trayectorias. La posibilidad de teledirigir un brazo articulado de un robot o de un robot humanoide, como es el caso, da la oportunidad de generar nuevas trayectorias aplicables al propio robot. Mediante el uso del procesamiento de imágenes, se creará la recogida de información del teleoperador para, más adelante, aplicar sobre el propio robot [1][7][8].

El objetivo de este artículo se enfoca en la implementación de una aplicación que sea capaz de capturar las características de movimiento del teleoperador a través del uso de una cámara Kinect y generar la información de cómo el robot teleoperado debe moverse para poder imitar el movimiento o trayectoria del teleoperador. Se busca que el movimiento se realice en tres dimensiones y que el robot reproduzca fielmente los movimientos realizados por la persona que se encuentra frente al dispositivo Kinect [3][4]. Para llegar a alcanzar dicho objetivo se dividió el proceso en una serie de subtareas.

La primera subtask está asociada con la detección de detectar una persona mediante el dispositivo Kinect y realizar la esqueletización de la misma. Después, se pasará al cálculo de los ángulos articulares necesarios para que el robot ejecute los movimientos. Por último, se pasarán los ángulos articulares calculados al simulador, mostrándolos en este último caso por pantalla y verificando la validez de estos.

A continuación el artículo se redactará de la siguiente manera. En el siguiente capítulo, se mostrará una breve idea del robot humanoide TEO sobre el que se ha probado los experimentos y la finalidad de ellos. En el tercer capítulo, se podrá ver como es necesario realizar una esqueletización del teleoperador. Después, en el capítulo 4 se explicará todo algoritmo del cálculo cinemático inverso a través del método RPY para el cálculo de los ángulos. Por último, se expondrán los experimentos y sus resultados, juntos con unas conclusiones sobre estos.

2. EL ROBOT HUMANOIDE CAMARERO TEO

En la Universidad Carlos III de Madrid, el grupo de robótica humanoide RoboticsLab[®], ha comenzado a desarrollar un robot humanoide móvil y autónomo TEO (Figura 1), el cual deberá insertarse como robot asistencial o robot personal en una oficina o un entorno de trabajo como un robot camarero. El componente principal de este robot que manipula objetos es su mano la cual lleva una bandeja incorporada.

TEO es la nueva versión que es capaz de caminar en cualquier dirección, sentarse en una silla, subir y bajar escaleras, o moverse por sí mismo en entornos humanos. Con un enfoque diferente, la altura, el peso y la rigidez de enlace se han incrementado. De hecho, pesa aproximadamente 65 kg, tiene una altura de 1,75 m, y tiene 28 grados de libertad. Además, los sistemas hardware y de control son mejores. Por lo tanto, se pueden manejar más información de los sensores y algoritmos de control complejos con una capacidad de cálculo superior. Los principales objetivos de esta investigación han sido la estabilidad del robot humanoide [5] y la generación de caminata o pasos [2][9].

El objetivo de esta investigación es indagar en el área de control con los robots humanoides con un enfoque en el diseño y ejecución de las habilidades de manipulación complejas. En particular, hay que abordar el control de la manipulación basado en el control de visión por ordenador.

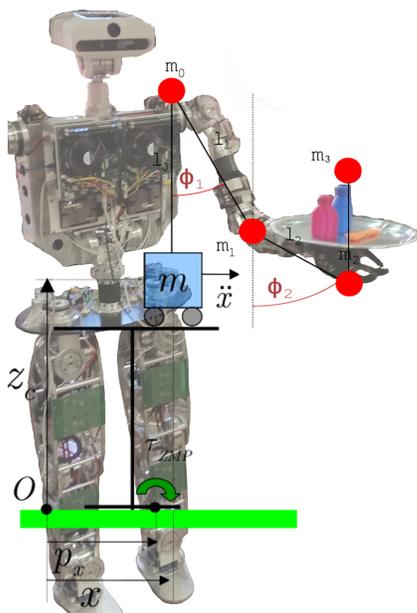


Figura 1: The waiter robot TEO.

3. ESQUELETIZACIÓN

El proceso de esqueletización consta en procesar las imágenes de profundidad obtenidas con Kinect para detectar formas humanas e identificar las partes del cuerpo del usuario, presente en la imagen. Cada parte del cuerpo es abstraída como una coordenada 3D o articulación. Un conjunto de articulaciones forman un esqueleto virtual para cada imagen 24 de profundidad de Kinect, es decir, se obtienen 30 esqueletos por segundo. Las articulaciones generadas varían de acuerdo a la biblioteca de Kinect que se utilice. Para este caso, cada esqueleto como lo indica la figura 18 está formado por 20 articulaciones $a_i = x_i, y_i, z_i$ con $z_i > 0$ cuyas coordenadas se encuentran expresadas en milímetros con respecto a la posición de Kinect en la escena [6][10][11][12].

La propia esqueletización se debe de encargar de asegurarse que el esqueleto rastreado no tiene partes fuera de la pantalla. Además, es capaz de rastrear de forma activa a dos esqueletos a la vez diferenciándolos y de forma pasiva a otros cuatro. Cada esqueleto será representado por un color diferente.

Los datos que se han obtenido son los puntos que representan partes representativas del cuerpo como centro de la cabeza, pecho o la cadera. Por lo tanto es necesario dibujar segmentos entre los puntos calculados y más tarde cada punto característico será resaltado en un color como se aprecia en la Figura 2.

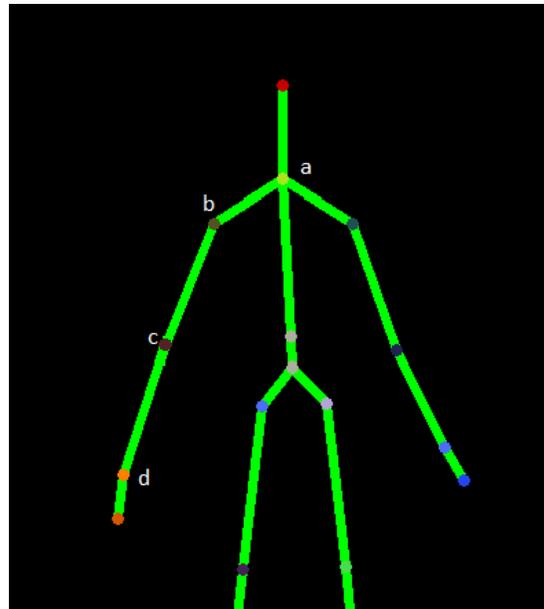


Figura 2: Esqueletización y marcación de las articulaciones características para la tarea de manipulación.

4. TELEOPERACIÓN BASADO EN RPY

Una vez terminado la subtask de la esquelización, el siguiente paso consiste en extraer la información que nos interesa para nuestro proyecto. En este caso se comenzará extrayendo la posición y movimiento del brazo derecho. Para esto se deberá obtener la cinemática inversa de la cadena cinemática compuesta por el codo y el hombro. De forma más concreta, se calcularán varios ángulos, siendo los más importantes el del codo (*angulo_codo*) y el formado entre el bíceps y el torso (*angulo_hombro*) y todo ello en un espacio tridimensional.

El primer paso será localizar los puntos característicos que se necesitan, en este caso serán la base del cuello, el hombro derecho, el codo derecho y la muñeca derecha. Se utilizará los datos del mapa del esqueleto ya preparados para salir por pantalla, por lo tanto se cuenta con que el centro de coordenadas está en la esquina superior izquierda y que no se cuenta con efecto espejo, por lo que el brazo de interés aparecerá en la parte izquierda. Todos los puntos estarán definidos sobre un espacio tridimensional.

Para simplificar la obtención de datos se supondrá los segmentos de la articulación en un plano imaginario. Para sacar el valor de cada ángulo se contará con la posición de los 3 vértices (base del cuello, hombro y codo para el *angulo_hombro* y el hombro, codo y muñeca para el *angulo_codo*). Con estos datos se puede sacar fácilmente los lados del triángulo permitiéndonos obtener los ángulos mediante cinemática inversa como se indica a continuación.

A continuación, se detalla el proceso a seguir los ángulos de rotación tanto de la articulación del hombro como del codo. Ya sea para el hombro o el codo, solo se describe en este artículo que proceso para definir el ángulo únicamente en el plano XY. Para la obtención de los grados en los planos XZ y YZ, los pasos a seguir serán exactamente los mismos pero utilizados los propios datos de cada plano a calcular. Para el caso del hombro contaremos con tres vértices formados por base del cuello, hombro y codo, los cuales forman un triángulo (ABC) del cual podremos sacar sus lados permitiéndonos así obtener los ángulos mediante el método RPY. Utilizando el teorema del coseno se obtiene el ángulo del hombro respecto al codo.

$$AH_{xy} = \arccos \left(\frac{BC_{xy}^2 + AB_{xy}^2 - AC_{xy}^2}{2 * BC_{xy} * AB_{xy}} \right) \quad (1)$$

Donde AH_{xy} es el ángulo entre las rectas AB y BC. AB_{xy} es la distancia entre los puntos A y B. BC_{xy} es la distancia entre los puntos B y C. AC_{xy} es la distancia entre los puntos C y A.

Por otro lado también es necesario calcular el ángulo del hombro respecto al cuello, para posteriormente restárselo. Hay que tener en cuenta también el ángulo formado por la recta que une la base del cuello y el hombro con la horizontal, ya que este no es fijo sino que va disminuyendo a medida que levantamos el brazo y si no lo tenemos en cuenta tendríamos un error en el ángulo del hombro bastante apreciable (Figura 3).

$$ACH_{xy} = \arctan \left(\frac{AB_y}{AB_x} \right) \quad (2)$$

Donde AB_x es la componente X de la recta AB. AB_y es la componente Y de la recta AB.

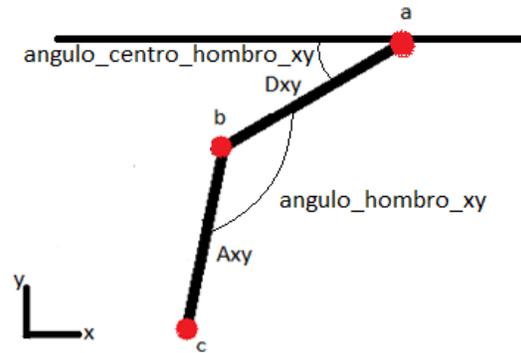


Figura 3: Esquema de la configuración y cálculo de ángulo del hombro respecto del cuello y del codo respecto al hombro.

Por último, para conseguir el valor final del ángulo de lo hombro, es necesario tener en cuenta el problema asociado de codo arriba/elbow up (EU) o codo abajo/elbow down (ED) para la articulación del hombro. Esta situación implica que dependiendo de si el codo se encuentra por encima o por debajo de la prolongación de la recta AB, el comportamiento del sistema será totalmente diferente. En la Figura 4, se muestra este inconveniente que influirá en los cálculos. De esta forma, teniendo en consideración las ecuaciones 1 y 2 y la posición del codo, se obtienen las ecuaciones que definen el ángulo del hombro:

$$EU \Rightarrow H_{xy} = \frac{3\pi}{2} - AH_{xy} - ACH_{xy} \quad (3)$$

$$ED \Rightarrow H_{xy} = AH_{xy} - \pi - ACH_{xy} \quad (4)$$

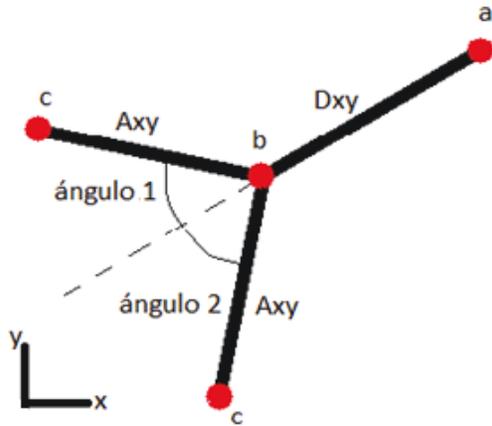


Figura 4: Esquema de la configuración codo arriba o codo abajo.

Una vez calculado el ángulo del hombro, pasamos a obtener el ángulo del codo. Para calcular los ángulos del codo contaremos con tres vértices formados por hombro, codo y muñeca (BCD), los cuales forman un triángulo del cual podremos sacar sus lados permitiéndonos así obtener los ángulos mediante el método RPY. Además hay que tener en cuenta que la muñeca no se ha considerado, por lo tanto, se simplificaran todos los cálculos asociados al codo. Utilizando el teorema del coseno se obtiene el ángulo del codo respecto a la muñeca.

$$AC_{xy} = \pi - \arccos \left(\frac{BC_{xy}^2 + CD_{xy}^2 - BD_{xy}^2}{2 * BC_{xy} * CD_{xy}} \right) \quad (5)$$

Donde AC_{xy} es el ángulo entre las rectas BC y CD. CD_{xy} es la distancia del punto C al D en el plano XY. BD_{xy} es la distancia del punto B al D en el plano XY.

5. EXPERIMENTOS Y RESULTADOS

Una vez explicado los pasos que a seguir para completar la aplicación, llega el momento de definir los experimentos y evaluar los resultados obtenidos para determinar la fiabilidad del programa. Para ello, aplicaremos todas las trayectorias sobre un simulador del robot humanoide TEO y de esa forma, comprobamos lo bien o mal que funciona el algoritmo. Para demostrar la eficacia del programa implementado, se ha realizado dos tiempos de pruebas. La primera está relacionada con la capacidad de obtener unos valores correctos durante el procesamiento de la imagen. Y el segundo está asociado con la velocidad de reacción ante movimientos rápidos por parte del teleoperador.

Por primer experimento se evaluarán los tres ángulos del hombro al mismo tiempo para comprobar que dichos ángulos se corresponden con la realidad. Para ello se empezará con el brazo pegado al tronco y se elevará de forma oblicua hasta que quede paralelo al suelo, es decir, que los ángulos de los planos XY e YZ están a unos 90°, momento en el que se moverá hacia adelante, hacia atrás, y de nuevo hacia adelante hasta quedar en la posición inicial paralelo al suelo, para posteriormente elevarlo pero sin llegar a forzar la articulación. La trayectoria de este movimiento se puede observar en la Figura 5.

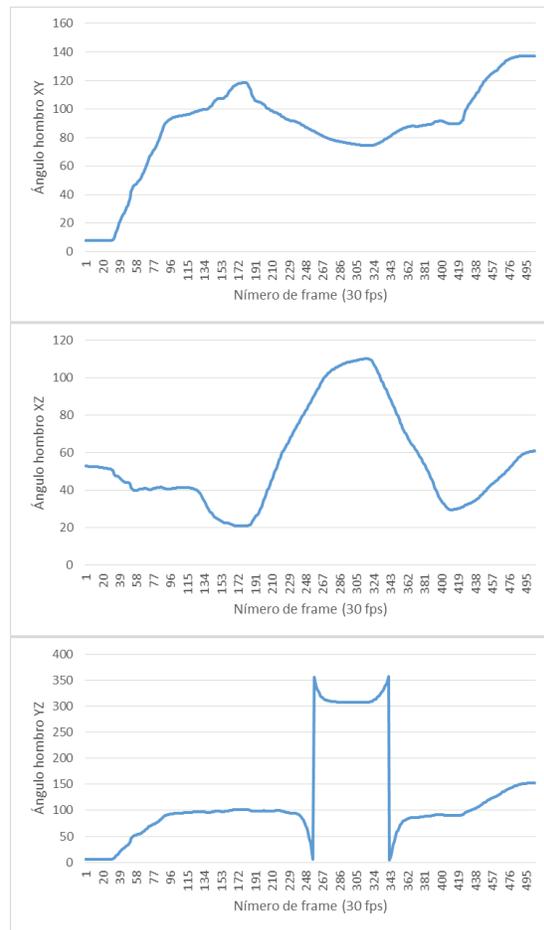


Figura 5: Ángulo del hombro en los planos XY, XZ y YZ para un movimiento compuesto.

Comparando las tres gráficas se puede apreciar el movimiento descrito anteriormente aunque con bastantes irregularidades ante la dificultad física de, por ejemplo, realizar movimientos paralelos al suelo manteniendo la misma altura. Fijándonos en la primera gráfica que describe el movimiento del hombro en el plano XY, se puede observar como partimos de casi 0°, es decir, el brazo pegado al tronco y como a medida que levantamos el brazo el ángulo va aumentando hasta alcanzar la posición deseada rondando los 90°.

Finalmente el ángulo vuelve a aumentar cuando levantamos el brazo para terminar el movimiento. Respecto a la segunda gráfica que representa el movimiento del hombro en el plano XZ, se puede observar que mientras elevamos el brazo de forma oblicua, el ángulo se mantiene próximo a los 45° , que sería el ángulo deseado. Posteriormente realizamos el movimiento hacia adelante con el brazo paralelo al suelo llegando hasta los 20° , luego hacia atrás llegando a los 110° y finalmente tratamos de volver a una posición de 45° aunque, como se aprecia en la gráfica, nos pasamos y llegamos hasta los 25° . Finalmente, al elevar el brazo tratamos de corregir la desviación anterior pasándonos esta vez y llegando hasta los 60° con el brazo ya levantado.

Por último, fijándonos en la última gráfica de la Figura 5 que representa el movimiento del hombro en el plano YZ, se ve como partimos de los 0° que representan el brazo pegado al torso y éstos van aumentando a medida que levantamos el brazo hasta estabilizarse en los 100° mientras realizamos los movimientos horizontales. Además, si comparamos la segunda y la tercera gráfica, se ve cómo en el momento en el que el ángulo del hombro en el plano XZ pasa de los 90° , es decir, el brazo pasa de estar por delante del torso a estar por detrás, el ángulo del hombro en el plano YZ pasa rápidamente de los 100° a casi los 310° . Si el movimiento horizontal fuese perfecto el ángulo debería pasar de 90° a 270° , sin embargo llegamos casi a los 310° debido a que los humanos al mover el brazo hacia atrás paralelo al suelo no podemos y tendemos a bajar el brazo para llegar lo más atrás posible. Cuando el brazo vuelve a estar por delante del torso el ángulo vuelve en este caso a los 90° y a partir de ahí vemos como a medida que levantamos el brazo en el último movimiento, el ángulo va subiendo hasta llegar a los 150° aproximadamente.

Para el segundo experimento, se ha tenido en cuenta los errores que han aparecido debido a la alta frecuencia de muestreo (30fps). Aunque en los experimentos anteriores se haya tomado una frecuencia muy alta, con la cual se es capaz de seguir movimientos muy bruscos, es necesario eliminar los errores producidos para movimiento bruscos. Para ello, se ha aplicado un filtro de paso bajo. Se ha mantenido las mismas condiciones que en el experimento anterior, y además se añade un filtro de paso bajo.

Como se observa en la Figura 6 y en primer lugar, el sistema es capaz de seguir el movimiento del brazo fielmente. Manteniendo esa velocidad de muestre de 30 fps, la respuesta del sistema es bastante robusta, pues es capaz de leer de manera

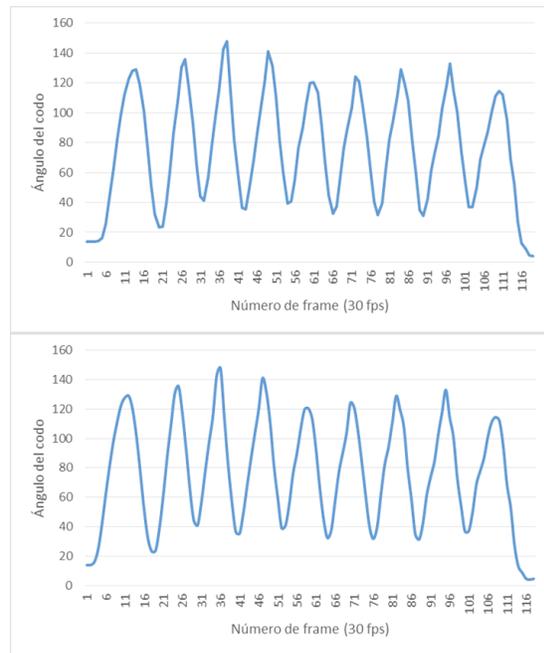


Figura 6: Ángulo del codo para movimientos de alta frecuencia con y sin un filtro paso bajo.

correcta el ángulo (en este caso el del codo) de manera precisa. Y en segundo lugar, además se aprecia que las anomalías o errores antes descritos se mitigan gracias al filtro paso bajo aplicado, dejando el movimiento mucho más constante y suave, y por tanto, más favorable para la posterior teleoperación del robot.

6. CONCLUSIONES

El desarrollo del proyecto ha servido para descubrir las enormes posibilidades que ofrece el sensor Kinect para llevar a cabo todo tipo de funciones relacionadas con la visión artificial. Sus capacidades no solo ofrecen buenos resultados a la hora de capturar a un usuario, también es capaz de funcionar como sensor de visión en cualquier robot que necesite captar y analizar su entorno. Esto es interesante, pues con la visión foveal que tienen los seres humanos, podemos bioinspirarnos para controlar los alrededores del robot y además interactuar con sus funciones de camarero.

En cuanto al sistema de teleoperación, el sistema se comporta de forma robusta en primer lugar ante cualquier tipo de movimiento ya sea en el plano XY, XZ o YZ. En cualquiera de los casos, se ha conseguido un seguimiento correcto. Y en segundo lugar ante movimientos con hasta velocidades de $20^\circ/s$, el sistema ha recodificado fielmente cada uno de los ángulos del brazo. Lo que permite asegurar que los datos sean fiables independientemente de la velocidad de acción de teleoperador.

Como trabajos futuros, el siguiente paso consistirá en desarrollar una cadena cinemática más compleja donde se tenga en consideración tanto el movimiento de la muñeca, como la consideración de la orientación del TCP (Tool Center Point). Esto permitirá dar más versatilidad a los movimientos. Otro paso a dar será la generación de nuevos movimientos más adecuados con las de un robot camarero. Esta base de datos permitirá elaborar a su vez comportamientos más complejos, los cuales darán naturalidad al comportamiento del robot.

Agradecimientos

The research leading to these results has received funding from the ROBOHEALTH-A project (DPI2013-47944-C4-1-R) funded by Spanish Ministry of Economy and Competitiveness and from the RoboCity2030-III-CM project (S2013/MIT-2748), funded by Programas de Actividades I+D en la Comunidad de Madrid and cofunded by Structural Funds of the EU.

Referencias

- [1] Alfaro Ballesteros, S. (2012). Sistema de teleoperación mediante una interfaz natural de usuario (Master's thesis).
- [2] Arbulú, M., and Balaguer C.. Real-Time Gait Planning for the Humanoid Robot Rh-1 Using the Local Axis Gait Algorithm. *International Journal of Humanoid Robotics*, 2009, Vol. 6.01, pp. 71-91.
- [3] Dai, X., Wu, M., Davidson, B., Mahoor, M., & Zhang, J. (2013, September). Image-based fall detection with human posture sequence modeling. In *Healthcare Informatics (ICHI)*, 2013 IEEE International Conference on (pp. 376-381). IEEE.
- [4] García, M. A., Cárdenas, A., Rendón, J. M., & Maya Méndez, M. (2009). Una Plataforma de Control Basado en Visión para la Rehabilitación de Robots Manipuladores de Tipo Industrial. *Computación y Sistemas*, 12(4), 409-420.
- [5] Kaynov, D., et al. A practical decoupled stabilizer for joint-position controlled humanoid robots. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2009)*, St. Louis, USA, 2009; IEEE, pp. 3392 - 3397.
- [6] Marcos Viforcós, E. (2012). Aplicación de las cámaras 3D al reconocimiento de actividades (Bachelor's thesis).
- [7] Montiel, O. P. (2008). Manipulador teleoperado inalámbricamente. Universidad de las Américas Puebla.
- [8] Ortega, E. N., & Villaluenga, L. B. (2004). Teleoperación: técnicas, aplicaciones, entorno sensorial y teleoperación inteligente. *Reports de recerca de l'Institut d'Organització i Control de Sistemes Industrials*, (5), 1.
- [9] Pardos, J.M., and Balaguer C.. Humanoid robot kinematics modeling using lie groups. *Climbing and Walking Robots*, Springer Berlin Heidelberg, 2005, pp. 569-575.
- [10] Plagemann, C., Ganapathi, V., Koller, D., & Thrun, S. (2010, May). Real-time identification and localization of body parts from depth images. In *Robotics and Automation (ICRA)*, 2010 IEEE International Conference on (pp. 3108-3113). IEEE.
- [11] Shotton, J., Sharp, T., Kipman, A., Fitzgibbon, A., Finocchio, M., Blake, A., & Moore, R. (2013). Real-time human pose recognition in parts from single depth images. *Communications of the ACM*, 56(1), 116-124.
- [12] Webster, D., & Celik, O. (2014, February). Experimental evaluation of Microsoft Kinect's accuracy and capture rate for stroke rehabilitation applications. In *Haptics Symposium (HAPTICS)*, 2014 IEEE (pp. 455-460). IEEE.