

Uso de la Realidad Aumentada (RA) como Herramienta Para la Enseñanza y el Aprendizaje de Manipuladores Robóticos.

Jairo Andrés Mantilla¹, Estefanía Gómez², William Pinto³

¹Grupo de investigación en dinámica, control y robótica, Escuela de ingeniería mecánica, Universidad industrial de Santander, Colombia. Email: jairo2218051@correo.uis.edu.co

²Grupo de investigación en dinámica, control y robótica, Escuela de ingeniería mecánica, Universidad industrial de Santander, Colombia. Email: estefania2190349@correo.uis.edu.co

³Grupo de investigación en dinámica, control y robótica, Escuela de ingeniería mecánica, Universidad industrial de Santander, Colombia. Email: wpintoh@uis.edu.co

Resumen

La Realidad Aumentada (RA) es una tecnología que proporciona experiencias interactivas al usuario superponiendo objetos virtuales en un ambiente real. En este trabajo se presenta una herramienta desarrollada usando realidad aumentada para el estudio y enseñanza de manipuladores robóticos, dirigida especialmente a universidades sin los recursos económicos para adquirir un manipulador serial. Fueron desarrollados tres programas que le permiten al estudiante visualizar y rectificar conceptos esenciales de la robótica como: transformación de sistemas coordenados, metodología de Denavit – Hartenberg, configuración de un manipulador robótico, entre otros. El aplicativo permite al estudiante interactuar de manera virtual con diferentes manipuladores, controlando mediante el uso del teclado las articulaciones del robot, y visualizando en pantalla en tiempo real distancias y parámetros respecto a un sistema coordenado fijo. Teniendo en cuenta que los manipuladores se encuentran renderizados, es posible realizar análisis de trayectoria y colisiones en un recorrido dado por el usuario. El aplicativo es fácilmente modificable, lo que permite la inclusión de nuevos manipuladores y el desarrollo de estrategias innovadoras.

Palabras clave: Realidad aumentada (RA); Manipuladores robóticos; Denavit – Hartenberg; Planeación de trayectoria.

Abstract

Augmented Reality (AR) is a technological tool that provides interactive experiences to the user by overlapping virtual objects in a real environment. This paper presents a tool developed using augmented reality to study and teach robotic manipulators, focused on universities without the economic resources to acquire a serial manipulator. Three programs were developed that allow the student to visualize and rectify essential concepts of robotics such as the transformation of coordinate systems, Denavit - Hartenberg methodology, and configuration of a robotic manipulator, among others. The application allows the student to interact virtually with different manipulators, control the robot's joints using the keyboard, and visualize on the screen in real-time distances and parameters concerning a fixed coordinate system. Considering that the manipulators are rendered, it is possible to perform trajectory and collision analysis on a path given by the user. The application is easily modifiable, allowing new manipulators to be included and developing didactic strategies for students.

Keywords: Augmented reality (AR); Robotic manipulators; Denavit – Hartenberg; Path planning

1. Introducción

Los ambientes de realidad aumentada (RA) proporcionan experiencias interactivas al usuario mezclando elementos virtuales (imágenes, sólidos, texto, entre otros.) en un ambiente real por medio de un dispositivo en tiempo real [1]. En el área de la

enseñanza les permite a los estudiantes experimentar un entorno de aprendizaje novedoso que facilita la asimilación de conceptos, dado que contribuye a que la educación sea experimental e interactiva al mismo tiempo. En un proyecto educativo, los estudiantes han considerado que el recurso promueve la calidad, imaginación y curiosidad de la persona, mejorando la

atención y motivación [2]. Adicionalmente permite una mayor flexibilidad al acceso de información gracias al uso de computadoras, donde es posible representar objetos que no pueden ser utilizados en ambientes educativos por limitaciones físicas y financieras [3].

El uso de la realidad aumentada puede dividirse en cuatro aplicaciones principales: medicina, planificación y control del movimiento de robots, interacción humano-robot y robótica de enjambres. En la robótica se contempla el diseño de trayectorias, la creación de escenarios de prueba y la interacción con objetos virtuales, permitiendo al usuario la operación segura, fácil y económica de manipuladores robóticos. Se encontró que la aplicación de la RA puede facilitar significativamente el proceso de programación y planificación de los movimientos de un robot, lo que reduce las horas de formación del usuario [4]. Pese a que pueden presentarse dificultades en la detección de marcadores debido a la luminosidad [5], la RA demuestra ser una herramienta importante para la robótica, a causa de que permite planear la trayectoria libre de colisiones en un ambiente tridimensional e incluso determinar la orientación del efector final [6] [7].

Con el ánimo de implementar procesos educativos económicos y accesibles usando la RA, se ha desarrollado una aplicación que permite al usuario interactuar con diferentes manipuladores seriales desplegándolos en un ambiente real. Se busca fortalecer conceptos fundamentales en la cinemática directa como los son: las transformaciones lineales, matrices de transformación homogénea, la metodología de Denavit-Hartenberg, tipos de juntas, entre otros. El aplicativo se concibe para ser utilizado especialmente en universidades e institutos educativos donde no se posea la capacidad económica para la adquisición de un manipulador serial comercial.

2. Metodología

La herramienta desarrollada busca afianzar los siguientes conceptos relacionados con el aprendizaje y operación de manipuladores robóticos.

2.1. Metodología de Denavit Hartenberg

Esta metodología permite conocer la ubicación de los sistemas de referencia de los eslabones de un manipulador robótico, esto se logra al definir las transformaciones relativas entre los sistemas coordenados consecutivos SC_{i-1} y SC_i , por medio de cuatro parámetros [8]. Estos parámetros explican que cualquier transformación entre sistemas coordenados Euclidianos puede ser expresada en cuatro movimientos básicos de la siguiente manera:

a) Una traslación de d unidades en la dirección Z_{i-1} .

- b) Una rotación de θ grados alrededor del eje Z_{i-1} .
- c) Una traslación de a unidades en la dirección X_i .
- d) Una rotación de α grados alrededor del eje X_i .

En donde i se refiere a la articulación actual, $i = \{1, 2, \dots, \#Articulaciones\}$.

La transformación entre sistemas coordenados consecutivos se puede expresar por una matriz con la misma expresión general [9]. Para obtener la transformación entre el sistema coordenado inercial y el del efector final para un manipulador robótico con n grados de libertad, se utiliza la siguiente expresión:

$$H_{0n} = H_{01} \cdot H_{12} \cdot \dots \cdot H_{(n-2)(n-1)} \cdot H_{(n-1)(n)} \quad (1)$$

La ecuación (1) permite calcular la transformación completa, la matriz H_{0n} contiene las operaciones de rotación y traslación relativas entre los sistemas coordenados. $H_{(i-1)(i)}$ es la matriz de Denavit – Hartenberg para los sistemas coordenados consecutivos de las articulaciones establecidas en la metodología.

2.2. Matriz de transformación homogénea.

Una matriz de transformación relaciona un sistema coordenado de referencia con otro específico, en este caso la transformación entre el sistema coordenado de la cámara $\{X_C, Y_C, Z_C\}$ y aquel situado en el centro del marcador $\{X_0, Y_0, Z_0\}$ (Figura 1). La matriz de transformación homogénea tiene la siguiente estructura:

$$H_{C0} = \left[\begin{array}{ccc|c} R & & & p_{xyz} \\ \hline 0 & 0 & 0 & 1 \end{array} \right] \quad (2)$$

La ecuación (2) presenta un tensor que se divide de manera general en dos partes, un arreglo R de dimensión 3×3 que guarda todas las operaciones de rotación y un vector columna p_{xyz} que contiene la traslación necesaria entre los sistemas coordenados en cada uno de los ejes de referencia [10]. A lo largo de todo el trabajo se utilizaron matrices de transformación homogénea que, según su aplicación fueron obtenidas de dos maneras: Aquellas que representan la transformación entre la cámara y el sistema coordenado del marcador (Figura 1) se calculan mediante el software Processing y para las matrices de transformación de los manipuladores robóticos se utiliza la metodología de Denavit – Hartenberg.

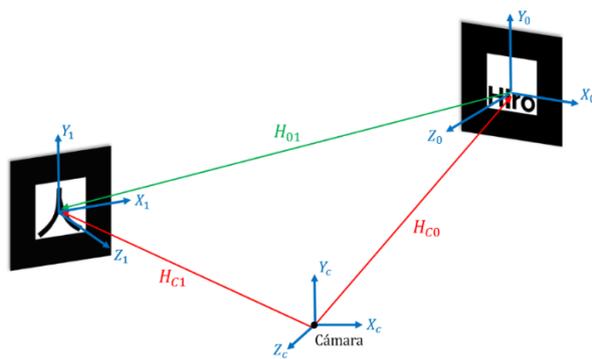


Figura 1. Sistemas coordenados y matrices de transformación entre la cámara y los marcadores de realidad aumentada. Fuente: Elaboración propia.

2.3. Realidad aumentada

Para exponer el aplicativo de realidad aumentada, se describen los componentes necesarios para su funcionamiento (Figura 2) y su operación de manera general. El objetivo de superponer objetos virtuales en un entorno real empieza con captar mediante una cámara la imagen del entorno donde se va a desarrollar la actividad, más adelante esta es sometida a un proceso de segmentación e identificación [11] en donde se reconoce la existencia de marcadores específicos que permite establecer las dimensiones del espacio. Una vez se cuenta con la identificación de los marcadores, según la aplicación seleccionada, se lleva a cabo el renderizado de los objetos virtuales sobre la imagen del entorno real en una pantalla o monitor.

En la figura 2 se muestran los componentes que fueron empleados para el desarrollo del aplicativo. Se hace uso de elementos que son fácilmente accesibles por el estudiante y que no representan un gasto económico significativo. La cámara que se utiliza puede estar integrada en la computadora y no requiere de una alta definición para su funcionamiento. El tamaño de los marcadores y la selección del obstáculo también queda a elección del estudiante.

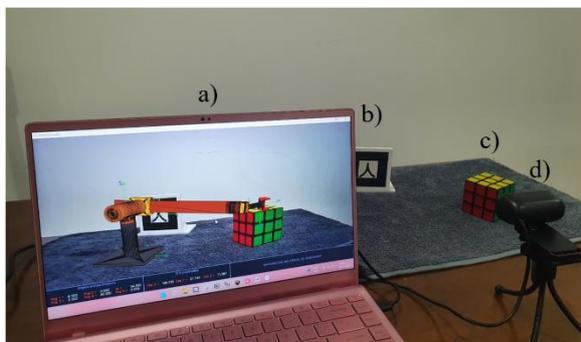


Figura 2. Componentes utilizados en la realidad aumentada. a) Computadora con ordenador. b) Marcador preestablecido Kanji.

c) Obstáculo. d) Cámara USB de alta definición, configurada en resolución de 640 x 360. Fuente: Elaboración propia.

2.4. Desarrollo del aplicativo

Para el desarrollo de los cuatro programas se presentan las siguientes etapas.

2.4.1. Modelamiento CAD

Fueron modelados tres manipuladores con diferentes grados de libertad (GDL). El segundo programa incluye: el robot SCARA y un robot propuesto por los autores. Para el tercer programa se implementa el robot STANFORD (Figura 3). Todos los componentes son exportados en formato tipo objeto 3D (.obj) para ser leídos por el software Processing.

Para obtener la matriz de transformación homogénea entre la cámara y el sistema coordenado de cada marcador se emplea la librería NyARToolkit 3.0.7 [12], adicionalmente las matrices de transformación entre la base y el actuador final de cada manipulador son desarrolladas con la metodología de Denavit – Hartenberg.

En la figura 4 se presenta el esquema funcional y el desarrollo de la metodología Denavit-Hartenberg para el manipulador propuesto de 4 GDL (Figura 3a). Se observan las dimensiones virtuales, los sistemas coordenados y la tabla de parámetros necesarios en la metodología. Este mismo proceso se realizó para los otros dos manipuladores.

$$X_1 = 140 * \text{Cos}(\theta_1 + 180) + 40 * \text{Cos}(\theta_1 + 180) * \text{Cos}(\theta_2) \quad (3)$$

$$Y_1 = 140 * \text{Sin}(\theta_1 + 180) + 40 * \text{Cos}(\theta_2) * \text{Sin}(\theta_1 + 180) \quad (4)$$

$$Z_1 = dz - 40 * \text{Sin}(\theta_2) + 30 \quad (5)$$

Las ecuaciones (3), (4) y (5) presentan las coordenadas del efector final respecto a la base del manipulador, con estas, se calculan las transformaciones de los efectores respecto a la cámara incluyendo en el análisis las matrices presentadas en la figura 1.

2.4.2. Implementación de la realidad aumentada en processing

Para el desarrollo de los programas de RA se utiliza el software de código abierto Processing [13] en su versión 4.0b8, que trabaja con el lenguaje JavaScript [14]. Dentro del programa se ejecutan las siguientes funciones:

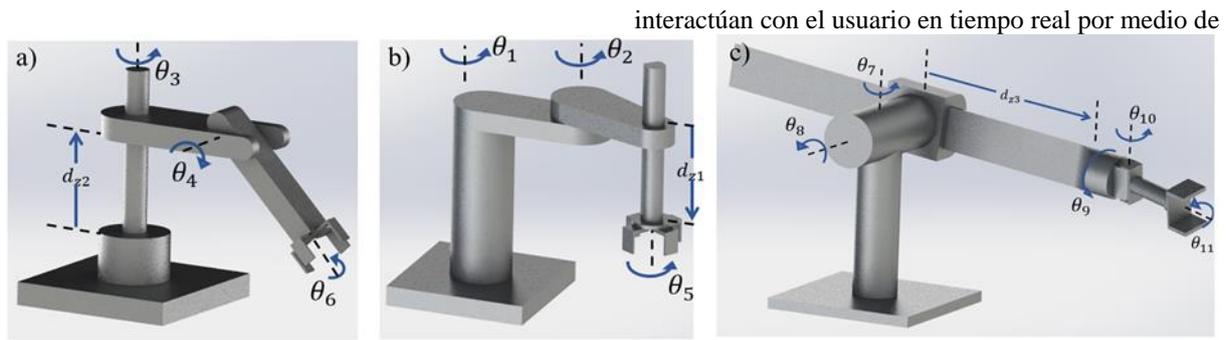


Figura 3. Manipuladores robóticos utilizados en el programa 2. a) Robot propuesto de 4 GDL. b) Robot SCARA de 4 GDL. C) Robot STANFORD de 6 GDL. Fuente: Elaboración propia

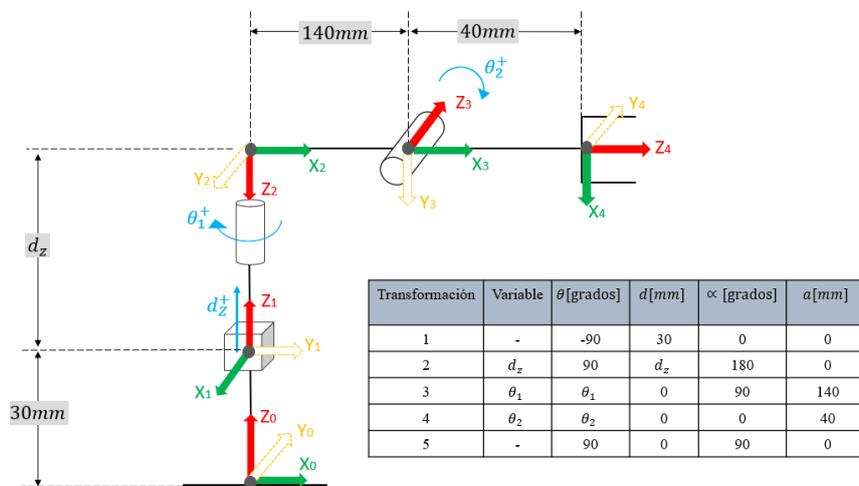


Figura 4. Esquema funcional del manipulador propuesto de 4 GDL

- Calibración de la cámara, este proceso es necesario para obtener información 3D a partir de imágenes 2D, existen diferentes técnicas para obtener los parámetros intrínsecos y extrínsecos de la cámara [15], fue implementado el Toolbox Single Camera Calibrator de MATLAB [16], el cual obtiene los parámetros por medio de una sesión de calibración, para el modelo de cámara implementado se obtienen los siguientes valores: distancias focales en pixeles $f_x = 523.34$, $f_y = 524.64$, punto principal $C_x = 304.69$, $C_y = 196.85$, coeficientes radiales de distorsión $k_1 = 0.3757$, $k_2 = 0.1167$ y los coeficientes tangenciales de distorsión $p_1 = -5.271 \times 10^{-4}$, $p_2 = 2.281 \times 10^{-4}$.
- Se utiliza la librería Processing video 2.0 [17] para enlazar la cámara, ejecutar las funciones de captura y renderizar la imagen.
- La realidad aumentada, la segmentación e identificación de los marcadores se ejecuta mediante la librería NyARToolkit 3.0.7 [12].

Se desarrollan cuatro programas complementarios para el aprendizaje de manipuladores seriales, todos estos

interactúan con el usuario en tiempo real por medio de

la posición de marcadores y la operación con el teclado.

Programa 1 “Sistema coordinado en marcadores”: En este aplicativo, usando la RA se encuentra la matriz de transformación homogénea entre la cámara y los marcadores. Se calculan las distancias y ángulos entre los marcadores tomando como referencia la cámara, estos valores calculados son visibles en tiempo real para el usuario en la parte inferior de la pantalla.

Programas 2 y 3 “Manipuladores seriales” y “Robot Stanford”: Estos programas están orientados para afianzar los conceptos de: tipos de articulaciones, geometría, espacio de trabajo y la metodología Denavit – Hartenberg. Usando los parámetros calculados en el primer programa se computan las distancias entre los efectores de los manipuladores en tiempo real, estas son visibles en la parte inferior de la pantalla. Se realizaron las matrices D-H para cada manipulador y los resultados se muestran en la parte inferior de la pantalla. El programa 2 presenta al robot propuesto y Scara, mientras que el 3 despliega al robot Stanford.

Programa 4 “Robot Stanford para el análisis de trayectoria”: Usando el manipulador del programa 3 el usuario controla el movimiento del robot por medio del teclado, al iniciar el análisis se coloca el manipulador en una posición inicial y se registra la trayectoria comandada por el usuario por medio de fotogramas que se toman con cierta frecuencia. Cuando el usuario lo prefiera se concluye el movimiento y se imprime en la pantalla el recorrido del objeto virtual que hace las veces del objeto que manipula el efector. Al mover la cámara, se tienen otras perspectivas de la trayectoria, adicionalmente es posible exportar los parámetros de las articulaciones durante la trayectoria.

3. Resultados

Se describe de manera detallada la participación que tiene el usuario con cada uno de los programas.

Programa 1 “Sistema coordinado en marcadores”: como se observa en la Figura 5, en este programa se visualizan los sistemas coordinados. En los marcadores el eje z es perpendicular a cada marcador y su origen coincide con el centro de la figura, el sistema coordinado de la cámara se imprime en la parte central de la pantalla. En la parte inferior de la pantalla se aprecian los parámetros de cada uno de los marcadores medidos respecto al sistema de referencia de la cámara (inercial), cualquier cambio respecto a la orientación o posición de los marcadores refleja un cambio en los parámetros. Adicionalmente se muestra la distancia entre los marcadores. Todos estos valores se pueden observar en tiempo real, el usuario puede interactuar con los marcadores para ver cómo cambian cada uno de los parámetros.

Programa 2 “Manipuladores seriales”: Es posible visualizar y operar los manipuladores Scara y propuesto. El usuario puede interactuar usando el teclado de su computadora para mover las articulaciones de cada uno de los robots, en la parte inferior de la pantalla podrá ver los parámetros de las articulaciones respecto al sistema de referencia que se encuentra en la base, es importante para el operador conocer la posición del efector final desde un marco de referencia fijo, en este caso los marcadores, por ello se imprimen las coordenadas, los ángulos y la distancia del mismo respecto a su marcador correspondiente

Programa 3 “Robot Stanford”: Presenta las mismas funciones que el programa anterior, en este caso solo se implementa el marcador “kanji” y el manipulador de 6 grados de libertad con el que puede interactuar el usuario.

Programa 4 “Robot Stanford para el análisis de trayectoria”: Usando el manipulador del programa tres, el usuario registra el movimiento desde un punto inicial a uno final que este disponga. Esta trayectoria se registra, es decir, se almacenan las variables de las articulaciones para cada configuración que toma el robot y una vez finalizado el movimiento se proyecta el recorrido que realizó el objeto virtual que desplaza el efector. Para obtener diferentes perspectivas es posible mover la cámara, esto permite determinar si la trayectoria se realizó libre de obstáculos o colisiones indeseadas como se presenta en la figura 8, adicionalmente se muestra el espacio de trabajo de este manipulador.



Figura 5. Programa 1, “Sistema coordinado en marcadores”. Fuente: Elaboración propia.

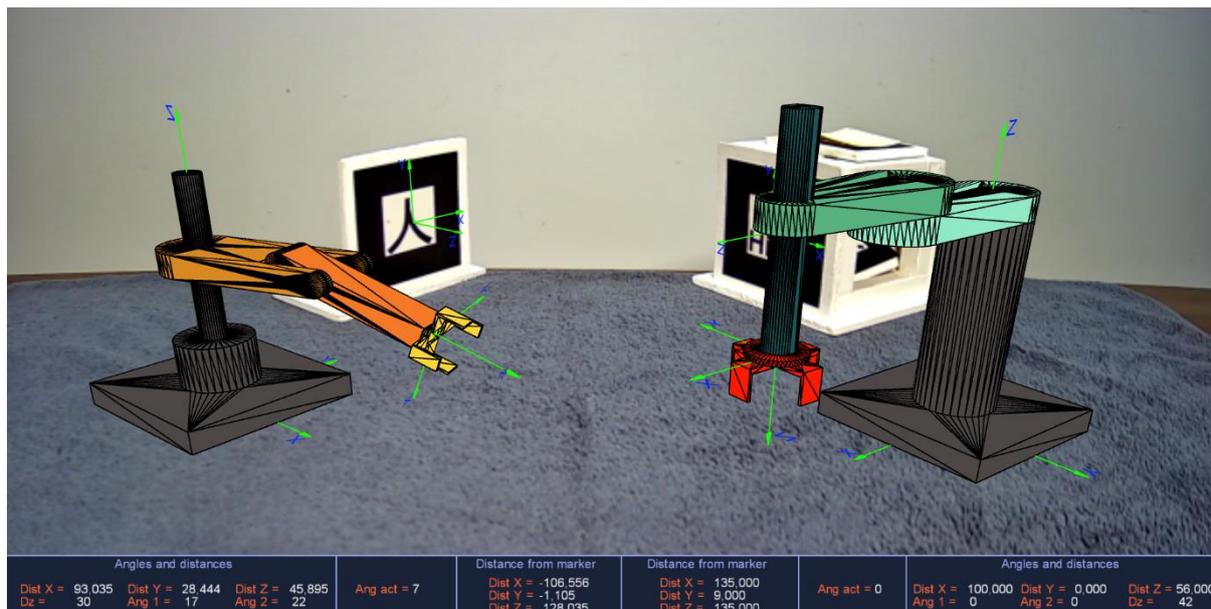


Figura 6. Programa 2 “Manipuladores seriales” de izquierda a derecha se imprime: ángulos y distancias de las articulaciones, distancias del efector final respecto al marcador. Se presentan los mismos datos para el siguiente manipulador. Fuente: Elaboración propia.



Figura 7. Programa 3 “Robot Stanford”. Fuente: Elaboración propia.

Es importante destacar que cualquier manipulador robótico puede ser modelado e incorporado en el aplicativo realizando pequeños cambios en su programación.

4. Conclusiones

En los diferentes programas expuestos el estudiante fomenta el aprendizaje de aplicaciones de la realidad

aumentada, transformación de sistemas coordenados, manipuladores robóticos, tipos de articulación, grados de libertad, espacio de trabajo y planeación de trayectoria.

La actualización instantánea de los parámetros medidos por la cámara, la configuración de los manipuladores y las coordenadas de sus efectores, hacen de la realidad aumentada una tecnología interactiva de aprendizaje. Al renderizar los

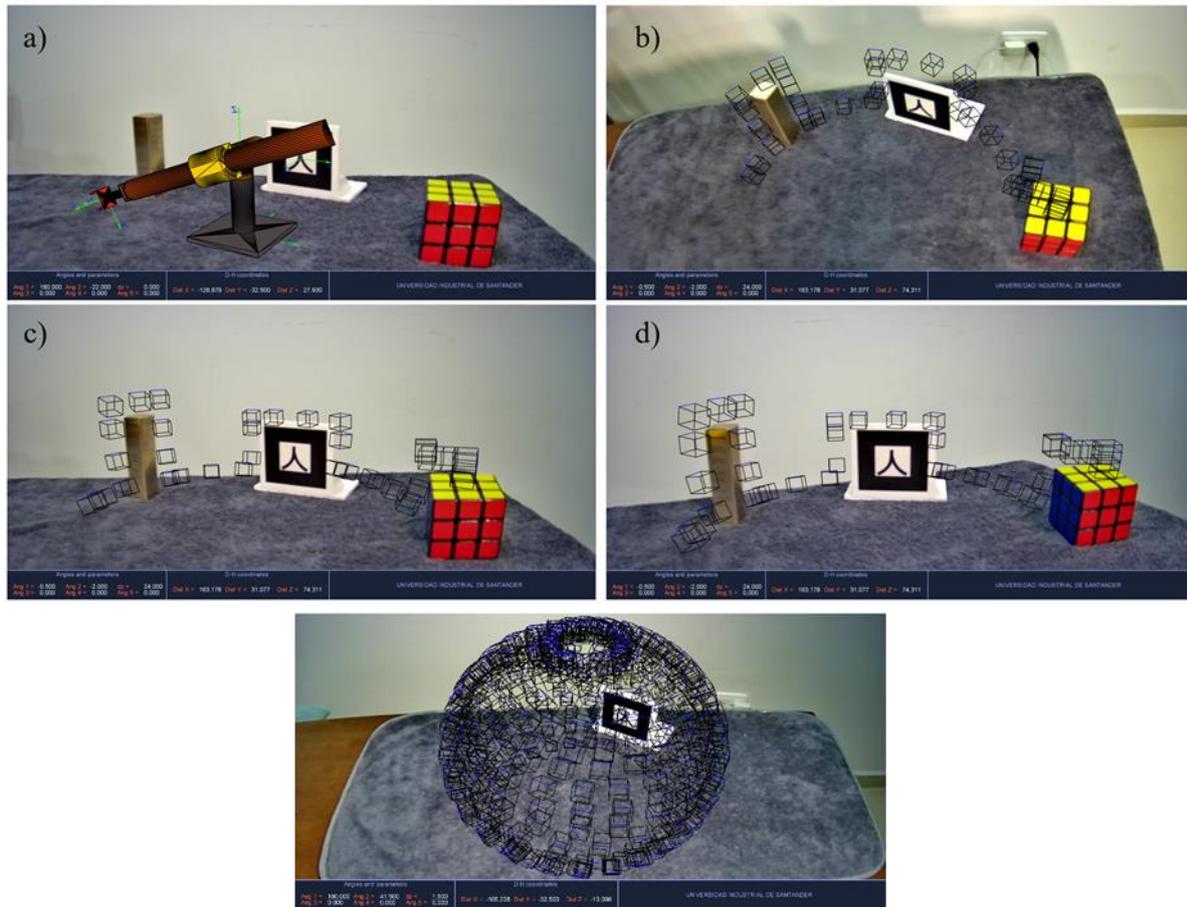


Figura 8. Programa 4 “Robot Stanford para el análisis de trayectoria”. Ejemplo de trayectoria libre de colisiones: a) Configuración inicial del manipulador, b) Vista frontal de la trayectoria, c) Vista lateral, d) Vista superior, e) Espacio de trabajo. Fuente: Elaboración propia

manipuladores robóticos en un ambiente real se tiene grandes ventajas para el análisis de trayectoria, permitiendo verificar de manera directa que no se generan colisiones o trayectos indeseados en el recorrido.

El desarrollo del aplicativo se presenta de manera que el usuario puede incorporar cualquier manipulador robótico con pequeños cambios en la programación, se plantea compartir el aplicativo para el acceso libre a personas interesadas en fortalecer los conceptos mencionado y mejorar la plataforma. El programa que corresponde al análisis de la trayectoria exporta los datos de cada articulación para cada posición almacenada, mismos datos que sirven de entrada en el desarrollo de procesos de control.

5. Agradecimientos

Agradecemos a la vicerrectoría de investigación y extensión (VIE), al decanato de ingenierías Fisicomecánicas y al semillero de robótica y control (DICBOT) por todo el apoyo brindado.

6. Referencias

- [1] Hernández, C. A. V., Martínez, A. R., & Ceballos, N. D. M. (2017). Caracterización de marcadores de realidad aumentada para su uso en robótica. *Revista Politécnica*, 13(25), 87-102. <https://doi.org/10.33571/rpolitec.v13n25a7>.
- [2] Cabero-Almenara, J., Vázquez-Cano, E., & López-Meneses, E. (2018). Uso de la realidad aumentada como recurso didáctico en la enseñanza universitaria. *Formación universitaria*, 11(1), 25-34. <https://dx.doi.org/10.4067/S0718-50062018000100025>.
- [3] Núñez, Y. J. D. (2014). Factibilidad tecnológica de aplicar realidad aumentada en la carrera ingeniería en ciencias informáticas. *3 c TIC: cuadernos de desarrollo aplicados a las TIC*, 3(4), 228-239.
- [4] Makhataeva, Z., & Varol, H. A. (2020). Augmented reality for robotics: A review. *Robotics*, 9(2), 21. <https://doi.org/10.3390/robotics9020021>.

- [5] Akçayır, M., & Akçayır, G. (2017). Advantages and challenges associated with augmented reality for education: A systematic review of the literature. *Educational Research Review*, 20, 1-11. <https://doi.org/10.1016/j.edurev.2016.11.002>.
- [6] Chong, J. W. S., Ong, S., Nee, A. Y., & Youcef-Youmi, K. B. (2009). Robot programming using augmented reality: An interactive method for planning collision-free paths. *Robotics and Computer-Integrated Manufacturing*, 25(3), 689-701. <https://doi.org/10.1016/j.rcim.2008.05.002>.
- [7] Fang, H. C., Ong, S. K., & Nee, A. Y. C. (2012). Robot path and end-effector orientation planning using augmented reality. *Procedia CIRP*, 3, 191-196. <https://doi.org/10.1016/j.procir.2012.07.034>.
- [8] Steinparz, F. X. (1985). Co-ordinate transformation and robot control with Denavit-Hartenberg matrices. *Journal of micro-computer applications*, 8(4), 303-316. [https://doi.org/10.1016/0745-7138\(85\)90031-4](https://doi.org/10.1016/0745-7138(85)90031-4).
- [9] Corke, P. I. (2007). A simple and systematic approach to assigning Denavit-Hartenberg parameters. *IEEE transactions on robotics*, 23(3), 590-594. <https://doi.org/10.1109/TRO.2007.896765>.
- [10] Pérez Cisneros, M. A., Cuevas, E., Zaldívar Navarro, D. (2014). *Fundamentos de robótica y mecatrónica con Matlab y Simulink* (1st ed.). Ra-Ma S.A. Editorial y Publicaciones.
- [11] Gupta, N., & Rohil, M. K. (2017). Exploring possible applications of augmented reality in education. In *2017 4th International Conference on Signal Processing and Integrated Networks (SPIN)*, 437-441. IEEE. <https://doi.org/10.1109/SPIN.2017.8049989>.
- [12] GitHub. (2016, April 13). NyARToolkit for Processing. <https://github.com/nyatla/NyARToolkit-for-Processing/blob/master/README.EN.md>. Processing Foundation (2022) Processing 4.0. <https://processing.org>.
- [13] Netscape Communications & Foundation Mozilla. (2021) JavaScript ECMAScript 202. <https://www.javascript.com>.
- [14] Viala, C. R., & Salmerón, A. J. S. (2008). Procedimiento completo para el calibrado de cámaras utilizando una plantilla plana. *Revista Iberoamericana de Automática e Informática Industrial RIAI*, 5(1), 93-101. [https://doi.org/10.1016/S1697-7912\(08\)70126-2](https://doi.org/10.1016/S1697-7912(08)70126-2).
- [15] The MathWorks, Inc. (2013). Camera Calibrator. <https://la.mathworks.com/help/vision/ref/cameralibrator-app.html>.
- [16] Processing Foundation (2022) Video. <https://processing.org/reference/libraries/video/index.html>.
- [17] Juarez-Salazar, R., Zheng, J., & Diaz-Ramirez, V. H. (2020). Distorted pinhole camera modeling and calibration. *Applied Optics*, 59(36), 11310-11318.