

TELE-OPERACIÓN DE UN BRAZO ROBOTICO UTILIZANDO REALIDAD VIRTUAL

REMOTE-OPERATION OF A ROBOTIC ARM USING VIRTUAL REALITY

Felipe Castro Aranza

Tecnológico Nacional de México / IT de Celaya, México
M20003075@itecelaya.edu.mx

Juan José Martínez Nolasco

Tecnológico Nacional de México / IT de Celaya, México
juan.martinez@itcelaya.edu.mx

Mauro Santoyo Mora

Tecnológico Nacional de México / IT de Celaya, México
mauro.santoyo@itcelaya.edu.mx

Recepción: 28/septiembre/2021

Aceptación: 30/noviembre/2021

Resumen

Se presenta el desarrollo de un ambiente de realidad virtual para la operación de un brazo robótico, donde se podrá visualizar un brazo robótico en 3D utilizando unos lentes de realidad virtual, el cual puede ser tele-operado con el teclado de la computadora.

Para mostrar el ambiente de realidad virtual se utiliza un headset HTC Vive Pro. La construcción del brazo robótico 3D se hizo con el software Blender, donde se crearon cada una de sus piezas además de un esqueleto con el cual delimitan sus movimientos. Para la integración de todos los elementos del ambiente de realidad virtual de tele-operación se recurrió al motor de videojuegos y realidad virtual de Unity, ya que actualmente es uno de los más utilizados con versión gratuita y con multitud de opciones para su programación. Así mismo se explica de forma general la exportación del modelo del brazo construido en 3D a la plataforma de Unity, la instalación del sistema de HTC (Visor, mandos y estaciones Base) y la programación de los movimientos del brazo robótico dentro del ambiente de realidad virtual.

Como resultado de este proyecto, se obtuvo la visualización del entorno creado desde los lentes de realidad virtual del sistema de HTC Vive Pro, y se logró dar movimiento al brazo robótico.

Abstract

The development of a virtual reality environment for the operation of a robotic arm is presented, where a robotic arm can be visualized in 3D using virtual reality glasses, which can be tele-operated with the computer keyboard.

To show the virtual reality environment, an HTC Vive Pro headset is used. The 3D robotic arm was built with the Blender software, where each of its parts were created, as well as a skeleton with which they delimited their movements. For the integration of all the elements of the tele-operation virtual reality environment, the Unity video game and virtual reality engine was used, since it is currently one of the most used with a free version and with a multitude of options for programming.

Likewise, the export of the arm model built in 3D to the Unity platform, the installation of the HTC system (Viewer, controls and Base stations) and the programming of the robotic arm movements within the reality environment are also explained. virtual.

As a result of this project, the visualization of the environment created from the virtual reality lenses of the HTC Vive Pro system was obtained, and it was possible to give movement to the robotic arm.

Keywords: *virtual reality environment, teleoperation, robotic arm*

1. Introducción

Con el rápido desarrollo tecnológico, las aplicaciones robóticas han penetrado en diversas áreas operativas en la industria de la construcción como vehículos aéreos no tripulados, para la construcción, escaneo de sitios, robots autónomos para la colocación de ladrillos [1-6] y robots humanoides en las operaciones industriales diarias [7]. Se observa que algunas operaciones en la industria se ven desafiadas por el entorno de trabajo abierto y en evolución, flujos de trabajo dinámicos y cambiantes [8,9]. En tal entorno, la teleoperación de robots es decir, trabajadores humanos para manipular sistemas robóticos en complejas tareas a distancia, es un

enfoque prometedor para converger las ventajas de sistemas robóticos y agentes humanos en condiciones inciertas para la toma de decisiones [10]. A diferencia de las soluciones totalmente autónomas que se basan en Inteligencia y detección robótica para la planificación y ejecución de tareas automatizadas [11], la teleoperación del robot permite al ser humano como comandante, una mejora en el tiempo de rendimiento y la calidad del trabajo, ya que puede escalar fuerza y movimiento para lograr capacidades de acción más fuertes, más grandes o pequeñas [12]. Debido a los importantes beneficios, la teleoperación de robots se ha implementado ampliamente en varios dominios con requisitos de trabajo complejos y potencialmente peligrosos para los seres humanos, como cirugías médicas [13-15], exploración espacial [16] y respuesta a desastres [17]. El origen de la realidad virtual es se remonta entre los años 1957 y 1962 con Sensorama [19]. En la misma época se empiezan a desarrollar algunos dispositivos con forma de casco que integran alguna pantalla como son el caso de Headsight, y Ultimate Display, los cuales son los predecesores a las gafas de realidad que se conocen hoy en día, dentro de las que destacan las desarrolladas en 1993 por SEGA. A partir de entonces y hasta hoy se han ido desarrollando softwares específicos y se ha ido mejorando la calidad y comodidad de los dispositivos, teniendo hasta ahora multitud de opciones como pueden ser Oculus Rift, Samsung Gear VR, HTC Vive Pro, entre otras.

La realidad virtual (RV) se utiliza en sectores profesionales como puede ser en la industria, en empresas para mejorar sus operaciones internas o incluso en los quirófanos y hospitales. Ejemplo de una aplicación en alguno de estos sectores es el sistema HoloSurg creado por Exovite, que utiliza las gafas HoloLens y realidad mixta para mejorar la seguridad y precisión en cirugías [referencia]. Otro ejemplo reciente del uso de la realidad virtual es en museos, como es el caso del museo Thyssen-Bornemisza, donde los visitantes pueden adentrarse y recorrer las escenas plasmadas en los cuadros de manera realista [22].

En la actualidad existen numerosos métodos de control para la teleoperación de robots, que van desde dispositivos móviles como tablets o teléfonos inteligentes hasta paneles de control con joysticks, volantes e incluso pedales. Estas interfaces

pueden llegar a resultar bastante complejas y aparatosas. Además, estas interfaces pueden presentar algunas limitaciones como son un campo de visión restringido, falta de orientación, percepción contextual y de profundidad incorrectas, imagen de video degradada y/o el periodo de latencia. Estas limitaciones, incluyendo efectos del factor humano como son una carga de trabajo excesiva, falta de conciencia de la situación, estrés, ansiedad y/o el exceso de confianza, pueden llegar a generar errores en tareas de teleoperación [21].

Con base en las restricciones de las tecnologías aplicadas en la teleoperación, es posible apreciar un área de oportunidad de la RV para la toma de decisiones de forma que la velocidad, precisión, grado de coordinación y carga de trabajo de los operadores se vea mejorada [21].

2. Métodos

El desarrollo del entorno de realidad virtual y su implementación con el equipo de realidad virtual HTC Vive, se basó en la metodología que se muestra en la figura 1.



Figura 1 Diagrama de la metodología.

Caracterización del ambiente de realidad virtual

Se definieron que los elementos necesarios dentro del ambiente de RV constarán de un brazo Fanuc LRMate200ic que deberá tener movimientos en sus 6 ejes como se muestra en la figura 2, conforme a esta demanda se utilizó el software de Blender para su diseño, esto ya que Bender permite además del diseño de objetos en 3D, la posibilidad de crear articulaciones para girar los objetos en las 3 direcciones X, Y y Z, con esto se facilitara permitir y restringir las rotaciones de los ejes.

Desarrollo de modelo de brazo robótico

Para la construcción del modelo en 3D, tomando como referencia las medidas y del brazo robótico Fanuc LRMate200iC utilizando las diferentes herramientas

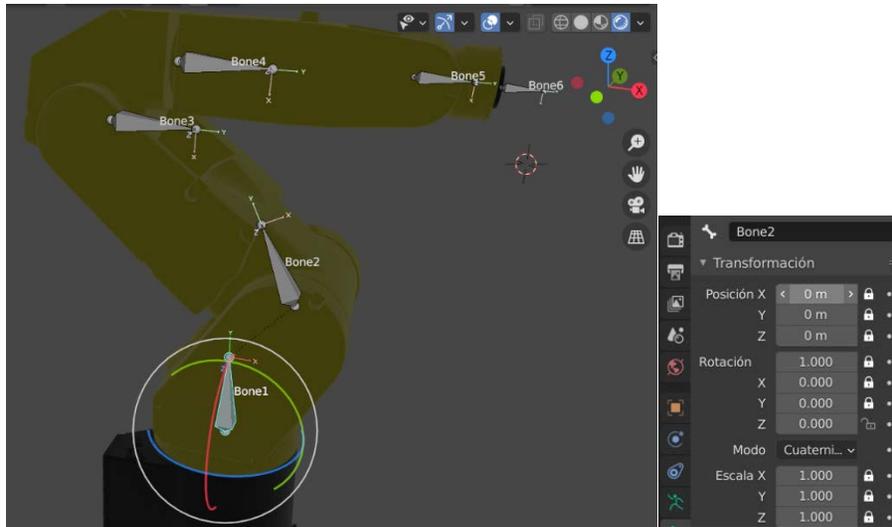


Figura 4 Vista del brazo robótico con articulaciones.

Tabla 1 Cuadro comparativo de motores de video juegos 3D.

Motores de Video juegos	HPL Engine 2	idTech 7	Unreal Engine 5	Unity 3D 2020.3
Lenguaje de scripting	C++	C++	VisualScript	C#, JavaScript, Boo
Plataformas de publicación	Windows, Mac OS, Linux	Microsoft Windows, PlayStation 4, Xbox One, Nintendo Switch	Playstation 3 y Xbox Series X/S	Windows PC, Windows, iOS, Android, Google Play, Amazon Appstore, SteamOS, Linux, PlayStation 3 y Xbox Series X/S, Samsung Smart TV, Android TV, PlayStation Vita, PlayStation 4, Xbox 360, Xbox One, Wii U, Nintendo 3DS, Nintendo Switch
Dispositivos de realidad virtual	No	No	HTC Vive y Oculus Rift	Oculus Rift, Google: Tizen, HTC Vive, Sony: PlayStation VR, Samsung Gear VR, Microsoft: HoloLens
API Grafica	OpenGL	OpenGL	DirectX	OpenGL, DirectX, Vulkan
Fisica	Newton's physics engine	PhysX	PhysX	PhysX
Importación de multiformato de modelos 3D	No	No	Si	Si
Control de Camara	Si	Si	Si	Si
Integración con navegadores web	No	No	No	Si
Editor grafico Integrado	Si	Si	Si	Si
Editor de terreno	No	No	Si	Si
Video juegos representativos	Estimote, Armada: The War Game	Quake 4, Wolfenstein	Fortnite, Halo Infinite, Apex Legends, Ark: Survival Evolved	Fortnite, Call of Duty: Warzone, Halo Infinite, Apex Legends

Se le dio principal atención a las características que facilitaban la integración de modelos en 3D, el diseño de la interfaz las plataformas de publicación, como Windows, Mac Os, Linux, Player web, el lenguaje de programación de scripting C++, JavaScript, Boo. De acuerdo con esta comparativa, el motor Unity resulto ser el más útil para el desarrollo de la interfaz (Figura 5).

Unity3D trabaja con programación orientada a objetos y con el lenguaje C#, para el desarrollo de este se utilizó Visual Studio. Para cada una de las mecánicas o los

procesos que no se podían realizar utilizando la herramienta de Unity. Se inicio primeramente programando el movimiento de cada una de las piezas las cuales serán controladas con el teclado. En la figura 6 se muestran algunas líneas de código para el movimiento de una pieza.

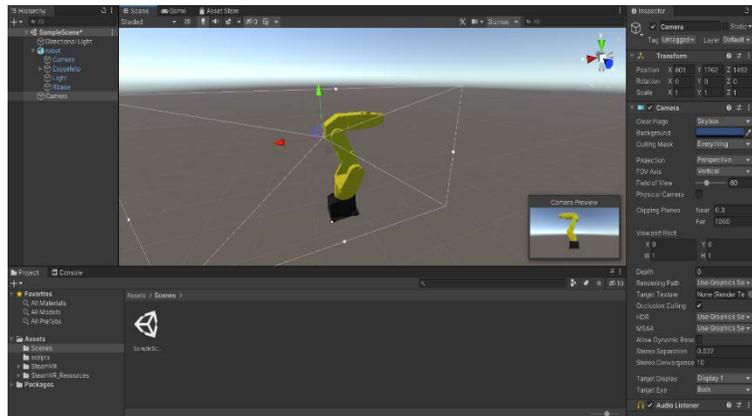


Figura 5 Vista del brazo robótico en Unity3D.

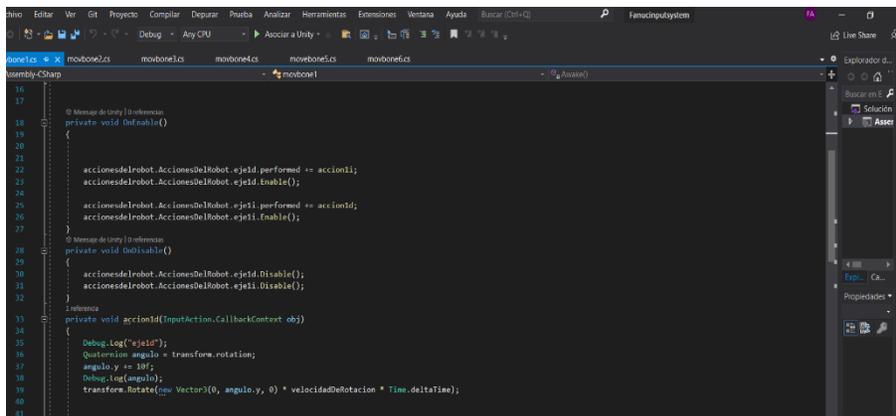


Figura 6 programación de movimiento en Visual Studio.

Definición de comandos de tele-operación

La entrada de datos es fundamental para que los proyectos en tiempo real sen interactivos. Una de las ventajas que ofrece el software Unity, es que dentro del entorno existe una herramienta llamada “Input System”, con esta herramienta se pueden implementar los controles de diferentes dispositivos como, por ejemplo: Gamepad, Joystick, Keyboard, Mouse, Touchscreen entre otros, solo se tiene que vincular las acciones de los dispositivos de entrada a la lógica del código.

Para la asignación de comandos, se declararon 12 botones del teclado como entradas, para posteriormente poder controlar las rotaciones positivas y negativas en los 6 ejes del robot (Figura 7).

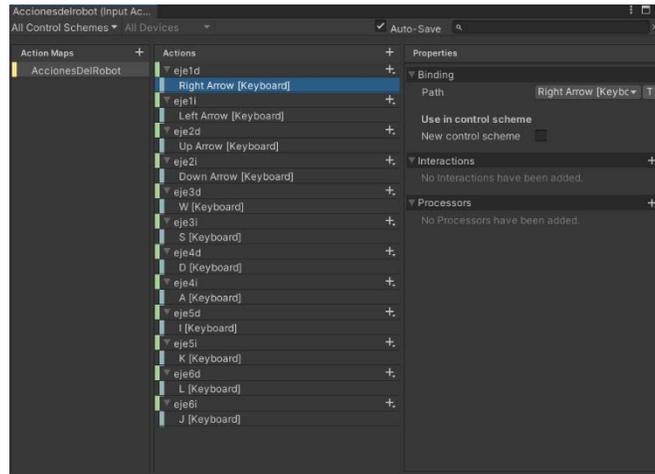


Figura 7 Input System, asignación de botones de entrada.

Definición de comandos de tele-operación

La entrada de datos es fundamental para que los proyectos en tiempo real sean interactivos. Una de las ventajas que ofrece el software Unity, es que dentro del entorno existe una herramienta llamada “Input System”, con esta herramienta se pueden implementar los controles de diferentes dispositivos como, por ejemplo: Gamepad, Joystick, Keyboard, Mouse, Touchscreen entre otros, solo se tiene que vincular las acciones de los dispositivos de entrada a la lógica del código.

Para la asignación de comandos, se declararon 12 botones del teclado como entradas, para posteriormente poder controlar las rotaciones positivas y negativas en los 6 ejes del robot (Figura 7).

3. Resultados

A continuación, se muestra la prueba del modelo que se desarrolló en la plataforma de Unity, se puede observar en la figura 8 al Usuario con el sistema HTC Vive, en la Figura 8 se puede apreciar el entorno de realidad virtual que operador ve a través de los lentes de realidad virtual, como primera etapa se lograron solo

movimientos lineales del brazo robótico con el teclado de la computadora, se continua trabajando para que el usuario con ayuda de los joysticks del sistema de HTC pueda elegir que parte del brazo desea mover y cuantos grados desea girarlo.



Figura 8 Usuario visualizando el entorno de realidad virtual con el sistema de HTC Vive.

Problemas encontrados

Durante la realización de este proyecto se han tenido que abordar diversos problemas. El primero fue la integración del headset de realidad virtual con la computadora, debido a que la tarjeta gráfica del equipo no era suficiente para soportar el equipo y poder ejecutar el software de configuración de Vive Pro. Por el lado del uso del software Blender surgieron errores típicos al utilizar un nuevo entorno de trabajo, al implementar la herramienta de Character Ringging se solucionaron tras buscar en su extensa documentación, figura 9.



Figura 9 Entorno que el usuario ve a través de los lentes de realidad virtual.

Dentro del motor de juegos Unity se tuvieron problemas con la programación del código para restringir los movimientos de los ejes y al momento de asignar acciones en el Input System que gracias a la amplia información que existe hoy en día se lograron resolver.

4. Conclusiones

Este proyecto fue desarrollado con la finalidad de elaborar una alternativa para la teleoperación de un brazo robótico, para ello se diseñó y se construyó este modelo basado en las medidas reales de un robot FANUC LR Mate 200iC.

Esta idea se puede implementarse para laboratorios virtuales donde cada alumno con su computadora y equipo de realidad virtual pueda realizar prácticas con un brazo robótico haciendo uso de esta plataforma.

Como trabajo futuro, se pueden incluir los sensores del sistema de HTC y así desplazar la cámara para que el usuario tenga diferentes perspectivas que más le acomode para el manejo de este, ya que como primera etapa solo se contempló solo una cámara para tener solo una vista del robot.

El objetivo principal de este proyecto era poder visualizar el brazo robótico fanuc desde los lentes HTC de realidad virtual, y poder ejercer el movimiento de sus piezas dentro del entorno de realidad virtual Unity y aprender el funcionamiento de los entornos y herramientas diseñadas para ello. Este objetivo se ha cumplido, logrando aprender a utilizar de forma básica un hardware y un software con el que no se había trabajado y experimentado antes. Se ha estudiado la documentación y algunos sistemas conocidos actualmente que han servido como base para alcanzar los objetivos propuestos. La investigación de este artículo servirá como base, a la hora de demostrar modelos de plataformas robóticas dentro de entornos de realidad virtual. Personalmente, elegí este trabajo por la curiosidad de conocer más a fondo un campo tan interesante en la actualidad como la realidad virtual y de aprender el manejo de una herramienta totalmente nueva y diferente. A pesar de los problemas encontrados este proyecto me ha servido para aprender el uso básico de herramientas como el sistema de HTC de realidad virtual y los programas de Unity y Blender que me podrán servir un futuro.

5. Bibliografía y Referencias

- [1] V. Vickranth, S. S. R. Bommareddy, and V. Premalatha, "Application of Lean Techniques, Enterprise Resource Planning and Artificial Intelligence in Construction Project Management," in *International Conference on Advances in Civil Engineering (ICACE-2019)*, 2019, vol. 21, p. 23.
- [2] F. Heintz, J. Kvarnstrom, P. Doherty, Bridging the sense-reasoning gap: DyKnow–stream-based middleware for knowledge processing, *Adv. Eng. Inf.* 24 (1) (2010) 14–26.
- [3] D. Hu, S. Li, J. Chen, V.R. Kamat, Detecting, locating, and characterizing voids in disaster rubble for search and rescue, *Adv. Eng. Inf.* 42 (2019) 100974.
- [4] F. Tian, M. Voskuijl, Automated generation of multiphysics simulation models to support multidisciplinary design optimization, *Adv. Eng. Inf.* 29 (4) (2015) 1110–1125.
- [5] D. Kim, A. Goyal, A. Newell, S. Lee, J. Deng, and V. R. Kamat, "Semantic Relation Detection between Construction Entities to Support Safe Human-Robot Collaboration in Construction," in *Computing in Civil Engineering 2019: Data, Sensing, and Analytics: American Society of Civil Engineers Reston, VA*, 2019, pp. 265-272.
- [6] M. J. Pivac and M. B. Wood, "Automated brick laying system for constructing a building from a plurality of bricks," ed: Google Patents, 2012.
- [7] H. Hasunuma et al., "A tele-operated humanoid robot drives a lift truck" in *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292)*, 2002, vol. 3: IEEE, pp. 2246-2252.
- [8] A. Dubois, L.-E. Gadde, The construction industry as a loosely coupled system: implications for productivity and innovation, *Construct. Management Economics* 20 (7) (2002) 621–631.
- [9] G. Hitz, A. Gotovos, M.-E. Garneau, C. Pradalier, A. Krause, and R. Y. Siegwart, "Fully autonomous focused exploration for robotic environmental monitoring," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, 2014: IEEE, pp. 2658-2664.

- [10] S. Cai, Z. Ma, M.J. Skibniewski, S. Bao, Construction automation and robotics for high-rise buildings over the past decades: A comprehensive review, *Adv. Eng. Inf.* 42 (2019) 100989.
- [11] P.F. Hokayem, M.W. Spong, Bilateral teleoperation: An historical survey, *Automatica* 42 (12) (2006) 2035–2057.
- [12] S. Hirche, M. Buss, Human-oriented control for haptic teleoperation, *Proc. IEEE* 100 (3) (2012) 623–647.
- [13] J. F. Jensen and J. W. Hill, “Surgical manipulator for a telerobotic system,” ed: Google Patents, 2003.
- [14] F. H. Moll et al., “Arm cart for telerobotic surgical system,” ed: Google Patents, 2005.
- [15] G.H. Ballantyne, Robotic surgery, telerobotic surgery, telepresence, and telementoring, *Surgical Endoscopy Other Interventional Techniques* 16 (10) (2002) 1389–1402.
- [16] D. Lester, H. Thronson, Human space exploration and human spaceflight: Latency and the cognitive scale of the universe, *Space Policy* 27 (2) (2011) 89–93.
- [17] J. Bohren, C. Paxton, R. Howarth, G. D. Hager, and L. L. Whitcomb, “Semi-autonomous telerobotic assembly over high-latency networks,” in 2016 11th ACM/ IEEE International Conference on Human-Robot Interaction (HRI), 2016: IEEE, pp. 149-156.
- [18] G. Guthart and J. Salisbury, “The intuitive telesurgery system: overview and application. W: Proceedings of 2000 IEEE International Conference; 2000, on Robots and Automation. San Francisco, 22-28 april 2000,” ed: IEEE, 2000
- [19] Olguin Carbajal, Mauricio; Rivera Zárata, Israel; Hernández Montañez, Erika *Introducción a la Realidad Virtual Polibits*, núm. 33, 2006, pp. 11-15 Instituto Politécnico Nacional Distrito Federal, México
- [20] Elena Peña Tapia (2017) “Interfaz inmersa para misiones roboticas basadas en realidad virtual”
- [21] HTC Corporation. (2016). Vive. 2020, abril 20: VIVE: <https://www.vive.com/eu/product/vive/>.

- [22] Tianyu Zhou, Qi Zhu, Jing Du.(2020) “Intuitive robot teleoperation for civil engineering operations with virtual reality and deep learning scene reconstruction”
- [23] Beatriz Salvador Ramos, Aplicación de realidad virtual para mostrar datos de monitorización en 3D capturados en entornos reales. Universidad de Zaragoza, 2019.
- [24] Carbajal, M. O., Zárate, I. R., & Montañez, E. H. (2006). Introducción a la Realidad Virtual. *Polibits*(33), 11-15. Recuperado el 20 de febrero de 2020, de <https://www.redalyc.org/pdf/4026/402640446002.pdf>
- [25] Digital Learning SL. (2015, 2 de noviembre). Scripts y lenguajes de programación en Unity. Academia Android: <https://academiaandroid.com/scripts-lenguajes-programacion-unity/>.
- [26] Eggberg, E. (2015). Valve and HTC Reveal Vive VR Headset. 2020, abril 20: GAMESPOT: <https://www.gamespot.com/articles/valve-and-htcreveal-vive-vr-headset/1100-6425606/>.
- [27] HTC Corporation. (2017). Vive Support. 2020, abril 20: VIVE. <https://www.vive.com/mx/support/vive/>.
- [28] Ruíz, F. [L_Draven]. (2016, octubre 28). FPS (First Person Camera) RAPIDO Y FACIL / FAST & EASY - UNITY: <https://www.youtube.com/watch?v=tLUDVn5V8fU>.
- [29] Unity Technologies. (2020). Quiénes somos. 2020, abril 20: Unity. <https://unity.com/es/our-company>.
- [30] Unity Technologies. (2020). Unity User Manual (2019.3). USA: Unity Documentation: <https://docs.unity3d.com/es/2019.3/Manual/UnityManual.html>.
- [31] SteamVr. (2020) Interaction System from The Lab. 12 de abril del 2020: https://valvesoftware.github.io/steamvr_unity_plugin/articles/Interaction-