

Observer Design for Augmented Reality-based Teleoperation of Soft Robots

García-Samartín, J. F.^{a,*}, López Pérez, I.^a, Yolcu, E.^b, del Cerro, J.^a, Barrientos, A.^a

^aCentro de Automática y Robótica, Universidad Politécnica de Madrid, c/ José Gutiérrez Abascal, n° 2, 28006, Madrid, España.

^bMiddle East Technical University, Dumlupınar Blv No 1, 06800 Cankaya, Ankara, Türkiye

Resumen

La realidad virtual y aumentada están ganando fuerza como herramientas de teleoperación de múltiples tipos de robots, tanto manipuladores como móviles, aunque no de robots blandos. Y es que, las dificultades inherentes a su modelado hacen muy difícil conjugar representaciones precisas y computacionalmente eficientes. En este trabajo, se presenta una interfaz de realidad aumentada para la teleoperación de estos dispositivos. El sistema desarrollado consta de las gafas Microsoft HoloLens 2 y de un computador central encargado de los cálculos. Su validación se lleva a cabo sobre PETER, un manipulador neumático altamente modular. A partir de los datos que recoge de los sensores, el computador estima, usando las físicas del programa de realidad virtual, la posición del robot. Los errores obtenidos son del orden del 5% de su longitud, lo que demuestra que el empleo de realidad aumentada no solo facilita la interacción de los operarios con manipuladores blandos, sino que puede insertarse en el bucle de control.

Palabras clave: Robótica, Sistemas Hombre-Máquina, Sistemas Mecatrónicos, Telerrobótica, Diseño de Control, Trabajo en entornos reales y virtuales, Observadores no lineales y diseño de filtros

Diseño de Observadores de para la Teleoperación con Realidad Aumentada de Robots Blandos

Abstract

Although virtual and augmented reality are gaining traction as teleoperation tools for various types of robots, including manipulators and mobile robots, they are not being used for soft robots. The inherent difficulties of modelling soft robots mean that combining accurate and computationally efficient representations is very challenging. This paper presents an augmented reality interface for teleoperating these devices. The developed system consists of Microsoft HoloLens 2 glasses and a central computer responsible for calculations. Validation is performed on PETER, a highly modular pneumatic manipulator. Using data collected from sensors, the computer estimates the robot's position based on the physics of the virtual reality programme. Errors obtained are on the order of 5% of the robot's length, demonstrating that augmented reality facilitates operator interaction with soft manipulators and can be integrated into the control loop.

Keywords: Robotics, Human Machine Systems, Mechatronic Systems, Telerobotics, Control Design, Work in real and virtual environments, Nonlinear observers and filter design

1. Introduction

In recent decades, numerous virtual (VR) or augmented reality (AR) interfaces for remotely operating robotic systems have been developed for diverse applications such as surgery (Wang et al., 2024), search and rescue (Cruz Ulloa et al., 2024), industrial manipulation (Gong et al., 2025), repairing tasks (Aschenbrenner et al., 2019) and rehabilitation (Makhataeva and Varol, 2020). This is largely due to the advantages they offer operators, such as more intuitive control,

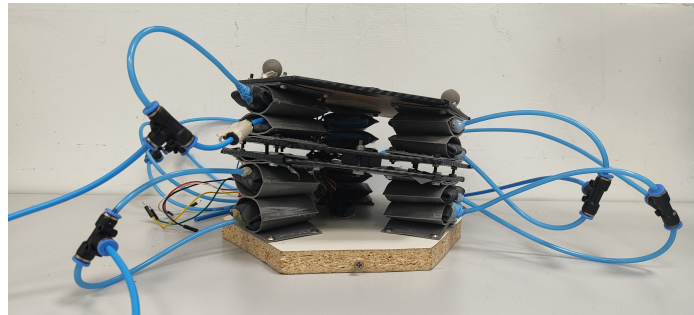
greater feedback, reduced cognitive load and enhanced safety, particularly in hazardous environments (Naceri et al., 2021; Su et al., 2023).

However, its applications in soft robotics are still very limited. Current simulation options in virtual reality software, coupled with the limitations of existing computing power, still prevent the faithful real-time reconstruction of the pose of these manipulators. This is because the information provided by the sensors is often indirect, requiring an observer to relate their readings to the robot's state.

*Corresponding Author: jorge.gsamartin@upm.es



(a) System Architecture.



(b) PETER Manipulator.

Figure 1: Developed system.

The first successful implementations of augmented or virtual reality in robots of this type relate to hyperredundant robots (Martín-Barrio et al., 2020). Soft robots can be considered an extreme example of this, with infinite modules. Other examples include mixed systems, such as the rehabilitation system presented in (He et al., 2025).

Three works stand out as particularly relevant in the field of soft robotics itself. Firstly, (Eslami et al., 2023) uses VR as an interface for controlling a silicone pneumatic finger. As it is a single-degree-of-freedom system with a curvature sensor, the reconstruction is straightforward and assumes constant curvature despite potential disturbances.

The same reconstruction philosophy is employed in (Borges et al., 2022) to model a Pneunet-type actuator in AR. However, greater accuracy is achieved by using six coloured markers as sensors, whose shape is reconstructed based on vertex transforms.

Finally, (Bern et al., 2024) uses a FEM-based algorithm to observe the state of a multi-degree-of-freedom cabled robot that is teleoperated with mixed reality. While the accuracy achieved is remarkable, the movement times are slow, raising the question of how much precision is necessary. Furthermore, the developed simulator appears to be highly specific to the robot in question, offering limited customisation options. Additionally, it does not address any aspects related to user experience in remote operation.

This work presents a new AR system for teleoperating a modular, parallel, six-input pneumatic platform. The system uses two inertial measurement units (IMUs) and two time-of-flight (TOF) sensors for measurement. These sensors send in-

formation to the AR system via a central computer, which then reconstructs the robot’s pose.

A robot of this type is ideal for remote operation using AR, as the information available for each module can be transferred to the virtual system. Furthermore, the relative simplicity of its actuation systems—particularly when compared to other pneumatic manipulators—makes it feasible to model them with a reasonable degree of accuracy and without an excessive computational load.

As in existing literature, therefore, the augmented reality system acts as the observer of the robot’s state. Special attention is given to the design of the user interface, and extensive options for robot customisation are provided.

Thus, it can be said that the work’s two main contributions are as follows:

- Firstly, the development of an augmented reality interface for teleoperating modular soft robots. Although some interfaces exist, they are very limited, and aspects such as customisation have not previously been addressed. This article presents a complete and intuitive interface.
- Secondly, the use of Unity as a state observer is explored. Although its precision will always be lower than that of FEM or NN-based solutions, this integrated solution has great potential since a virtual model of the robot must always be created and its accuracy evaluated.

The rest of the article is structured as follows: Section 2 describes the proposed system, which comprises the physical manipulator, the AR interface and the Unity model. Section 3

presents the state observer. Section 3 describes the results, and Section 5 draws the conclusions.

2. Augmented Reality System

2.1. System Overview

The developed AR system, whose architecture is depicted in Figure 1(a), aims to teleoperate pneumatic soft robots composed of modules with various actuators arranged in a parallel structure. To validate the system, the PETER manipulator, which was presented in previous work and consists of two modules, was used. Microsoft HoloLens 2 (Microsoft, 2019) headsets were employed and an interface called PETER-H was developed for these. The system model runs on a central computer and is called PETER-DK.

PETER-DK is responsible for receiving sensor information from the real robot, transmitting control commands back to it and indicating to the interface which model to display. PETER-DK communicates with the physical manipulator via a serial connection, while the central computer communicates with the headset via the TCP/UDP protocol.



(a) PETER-H Interface with PETER-DK on the back.



(b) PETER-H Menu. The available options in PETER-H menu are: toggle control sliders, edit geometric parameters, drag the PETER model, fix the PETER model and connect to PETER-DK.

Figure 2: PETER-DK and PETER-H.

2.2. PETER Manipulator

Figure 1(b) shows PETER, a pneumatic manipulator. Consisting of two modules, each of which is composed of three parallel actuators, it has five degrees of freedom and six air inlets. Each actuator has a TPU structure and inflatable rubber components, providing high load capacity and extensibility.

Each module incorporates a TOF and an IMU, enabling unique sensorisation. Construction details can be found in (García-Samartín et al., 2024).

Previous work has addressed the resolution of its kinematics (García-Samartín et al., 2025). However, it is difficult for operators who are not specialised in robotics to understand it properly, as well as to manage obstacles in the environment. This makes the development of interfaces like the one presented in this work highly advantageous.

2.3. PETER-H Interface

The second element of the system is the AR interface, which is represented in Figure 2(a). The central element of this is the robot's visualisation. Alongside this are sliders for setting the desired configuration and buttons for indicating that PETER should be moved to that position. There is also a menu, depicted in Figure 2(b), for managing the connection with the central computer and customising the manipulator's geometric parameters.

This subsystem operates sequentially, as described in the state diagram of Figure ???. Once connected to PETER-DK, the interface enables users to configure the robot's various geometric options, as detailed in Section 2.4. A virtual robot is then generated, passing from State 0 to State 1 that the user can drag and rotate freely within their field of view until the lock button is pressed, in which system enters in State 2. One of the platforms can then be selected, and its position adjusted using the sliders. At this point, the operator can ask the physical PETER to move to the specified position and enter State 3. A PID will then be activated until PETER reaches its position with an error of 3 mm or less.

2.4. PETER-DK Server

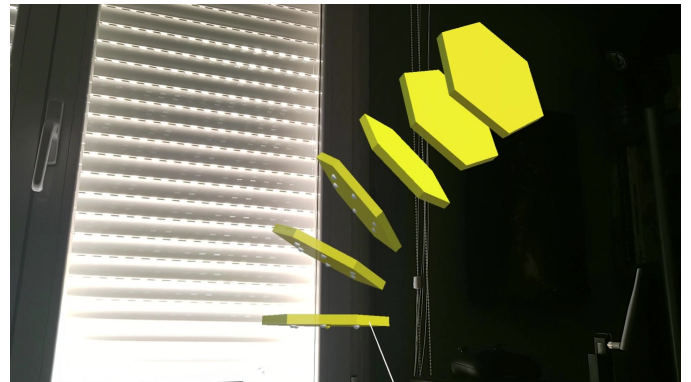


Figure 4: AR representation of PETER with 5 modules, each with 3 actuators.

This subsystem is responsible for receiving and processing sensor data for visualisation and control purposes, and for providing PETER-H with the necessary visualisation instructions.

PETER-DK was developed entirely in Unity. One of the objectives in the design phase was to make it suitable for the broadest range of parallel soft robots possible. To achieve this, the module was established as the basic unit of calculation, meaning the system estimates the kinematics of each module sequentially before concatenating them.

The user can define various parameters for each module, including the number of actuators, the radius of the polygon whose vertices are the actuators, the platform height, and the maximum and minimum elongation of the actuators. These parameters are calculated to ensure safety and prevent the physical manipulator from being sent with excessively inflated actuators that could cause leaks or breakage.

Figure 4 shows an example of a virtual robot with multiple modules. PETER-DK has been demonstrated to calculate the kinematics of 9 modules in real time, showcasing its capabilities. However, this does not mean that the system as a whole can function with such a large number of modules, as managing communications would likely result in high latency.

3. Observer Design

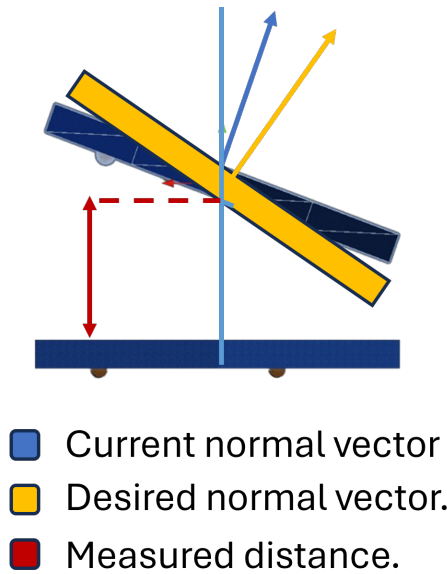


Figure 5: Model assumptions: rotation over a fixed point and height equal to the average height of the actuators.

In addition to designing an AR-based teleoperation interface for soft robots, this work aims to study the interface's models and their ability to predict the state of soft robots. Therefore, when configuring the model, it is essential to establish geometric and physical hypotheses that are consistent with reality but also simple enough to allow for real-time execution.

From the outset, therefore, the pneumatic nature of the system was disregarded and the actuators were considered as extendable rods. The inflation commands issued by PETER-DK to the physical manipulator correspond to the lengths calculated by the central computer, as characterised in García-Samartín et al. (2025). This simplification, based on good results obtained in previous work, significantly reduced computational cost with no noticeable loss of precision.

Unity's physics determined the lengths necessary for the central point of the module to reach a point (x, y, z) . These configurations were based on the assumption that the upper platform of each module rotated around its fixed central point and that this central point was always at an average height equal

to that of the actuators. Unlike the usual assumption in robotics of constant curvature, this hypothesis simplifies the calculations and is valid because the length of the actuators is similar to their separation, and no module reaches curvatures greater than 10° . Figure 5 shows a representative diagram of these assumptions.

For each module, PETER-DK receives the height value measured by the TOF sensor, h , as well as the IMU's rotation values around the X and Y axes, denoted by φ and ϑ , respectively. The program processes these values and informs Unity of the lengths of the actuators l_1, l_2 and l_3 .

Assuming each module has 3 actuators forming a triangle with side L , the positions of its bases are the ones specified in Equation (1):

$$\begin{aligned} \mathbf{a}_0 &= [L, 0, 0]^T, \\ \mathbf{b}_0 &= \left[-\frac{L}{2}, \frac{\sqrt{3}L}{2}, 0 \right]^T, \\ \mathbf{c}_0 &= \left[-\frac{L}{2}, -\frac{\sqrt{3}L}{2}, 0 \right]^T \end{aligned} \quad (1)$$

Rotation matrix of module i with respect to module $i-1$ can be obtained directly from the readings of the IMU by considering a rotation φ around the X-axis and a rotation ϑ around the Y-axis, as illustrated in Equation (2):

$${}^{i-1}\mathbf{R}_i = \mathbf{R}_x(\varphi)\mathbf{R}_y(\vartheta) = \begin{bmatrix} \cos \varphi & \sin \varphi \sin \vartheta & \sin \varphi \cos \vartheta \\ 0 & \cos \vartheta & -\sin \vartheta \\ -\sin \varphi & \cos \varphi \sin \vartheta & \cos \varphi \cos \vartheta \end{bmatrix} \quad (2)$$

The position of the end of each actuator on the upper platform is then calculated by applying the rotation matrix to its lower end and adding a displacement on the Z axis, as shown in Equation (3):

$$\begin{aligned} \mathbf{a}_i &= {}^{i-1}\mathbf{R}_i \mathbf{a}_{i-1} + [0, 0, h]^T \\ \mathbf{b}_i &= {}^{i-1}\mathbf{R}_i \mathbf{b}_{i-1} + [0, 0, h]^T \\ \mathbf{c}_i &= {}^{i-1}\mathbf{R}_i \mathbf{c}_{i-1} + [0, 0, h]^T \end{aligned} \quad (3)$$

PETER-DK uses this information to calculate the length of the actuators l_i , the position of the centre of the upper platform, \mathbf{x}_i , as well as its normal vector, \mathbf{n} . This parameter is necessary in order to determine the position of the centre of the lower platform of the next module, \mathbf{x}_{i+1}^{base} . As the upper platform is rigidly connected to the lower one, with their centres separated by a distance d , the calculation is performed using the Equation (4):

$$\mathbf{x}_{i+1}^{base} = \mathbf{x}_i + d \mathbf{n}' \quad (4)$$

If Rotation matrix of module i in global axis, ${}^0\mathbf{R}_i$ is desired, it can be obtained multiplying the matrix in local axis by all the previous rotation matrix, as Equation (5) presents:

$${}^0\mathbf{R}_i = {}^0\mathbf{R}_1 {}^1\mathbf{R}_2 \dots {}^{i-1}\mathbf{R}_i \quad (5)$$

Although the IMU measurements were relatively stable, the TOF exhibited significant noise in the form of sharp peaks that occurred randomly every few samples. To combat this, it was decided to filter the measurements. Although a mean filter seemed the most obvious choice, it was found that outliers excessively altered the result unless a very large number of measurements were used. This resulted in a delay that was too high

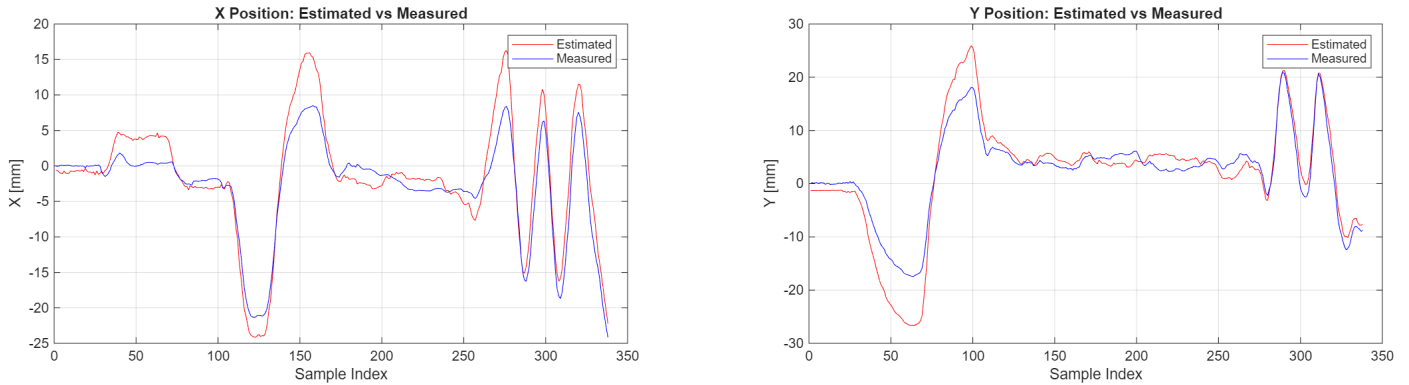


Figure 6: Comparison between the observed position (in red) and the real position (in blue) along the X and Y axis. Distances are displayed in mm.

Table 1: Error Statistical Analysis. Errors are displayed in mm.

Metric	MAE	RSME	Std. Dev.	Max. MAE	Q1 (MAE)	Q3 (MAE)
X axis	2.41	3.04	2.78	7.87	0.98	3.61
Y axis	2.59	3.56	3.55	9.35	1.09	2.96
Global	3.93	4.68	2.54	10.19	2.04	5.37

for the control requirements. Therefore, a Kalman filter was fitted experimentally. Values of $Q = 0.01$ and $R = 0.40$ were used, which indicated much greater confidence in the model than in the TOF measurements.

4. Results

Finally, the performance of the various subsystems and the complete system was evaluated. The AR interface and its characteristics of usability, adaptability and comfort have not been studied in depth and remain a future area of research. The development of a complete extended reality system for soft robots has been considered a commendable achievement in itself, given the scarcity of existing solutions.

However, to verify these systems' validity as state observers, the error between the position estimates provided by PETER-DK and those achieved by PETER in reality was analysed. A 15-camera OptiTrack system with an accuracy of around one-tenth of a millimetre was used as the reference standard.

During the test, the robot followed a trajectory for one minute. Every 100 ms, the observer described in Section 3 estimated the robot's position, while the Optitrack system captured the actual position. A total of 339 samples were obtained and distributed throughout the robot's workspace. Figure 6 shows the trajectories along the X and Y axes.

As depicted, the trajectory estimated by the robot is very similar to the actual one, particularly at the central points. When the robot reaches the far ends of its workspace, the system tends to overestimate the robot's position.

The same phenomenon can be observed in Table 1, where the difference between MAE and RMSE is approximately 0.7 mm. This indicates the presence of large point errors of up to 10 mm, which significantly increase the overall error. However, considering that the combined length of the two modules is 85 mm, the results suggest that the observer makes errors of 4.6% and 5.5%, respectively. These values are very good for Soft

Robotics applications. While they do not match the 2% error margin achieved in Bern et al. (2024), this result is obtained at a lower computational cost with greater integration capacity with AR.

At first glance, studying the two axes separately may suggest that the errors are similar. However, the standard deviation of the Y-axis is higher than that of the X-axis. Similarly, the third quartile of errors on the Y-axis is half a millimetre lower than on the X-axis, meaning its estimates are more precise, but its outliers are larger.

Therefore, to improve the observer's accuracy, their estimates must be refined in the areas of greatest error. As mentioned, these areas coincide with the boundary of the workspace. This may be due to the linearity assumptions being less accurate the further the robot moves from its central position, or because dynamic effects occur in these areas, which are reached at higher speeds and are not taken into account by the observer. While the former issue can be resolved by applying a correction factor or by slightly relaxing the model's assumptions, albeit at the expense of increased computational complexity, the latter issue would necessitate a more in-depth study involving the application of more advanced simulation techniques.

5. Conclusions

This paper presents an augmented reality interface for the teleoperation of modular soft robots. While common in many areas of robotics, these interfaces are underdeveloped in soft robotics. This is primarily due to the difficulty of developing observers that are both fast and accurate.

This problem has been addressed in this article. Using Unity as the development software, a dual glasses-central computer system has been implemented. The glasses act as the interface, while the computer handles calculations and communication with the actual robot. When receiving data from the robot's sensors, the system uses its own physics, with

some assumptions, to represent the robot's position. The results show errors of 5% of the robot's length, very good accuracy compared to other developments in the literature, although performance could be improved in the boundary areas of the workspace. Furthermore, a future direction is proposed: validating the AR interface with real users to confirm its functionality to introduce improvements in its ease of use.

Acknowledgments

This work is the result of research activities carried out at the Centre for Automation and Robotics, CAR (UPM-CSIC), in the facilities of the Escuela Técnica Superior de Ingenieros Industriales, within the Robotics and Cybernetics research group (RobCib). This work is supported by "Ayudas para contratos predoctorales para la realización del doctorado con mención internacional en sus escuelas, facultad, centros e institutos de I+D+i", which is funded by "Programa Propio I+D+i 2022 from Universidad Politécnica de Madrid", by the "Proyecto CollaborativE Search And Rescue robots (CESAR)" (PID2022-142129OB-I00) funded by MCIN/AEI/10.13039/501100011033 and "ERDF/EU", and by the "iRoboCity2030-CM" (EC-2024/TEC-62) project, funded by the "Dirección General de Investigación e Innovación Tecnológica de la Comunidad de Madrid" pursuant to Order 5696/2024.

Annex: Code

The code of the two developed subsystems can be found in the following links:

- <https://github.com/Robcib-GIT/PETER-H>
- <https://github.com/Robcib-GIT/PETER-DK>

References

Aschenbrenner, D., Leutert, F., Çençen, A., Verlinden, J., Schilling, K., Latoschik, M., Lukosch, S., 2019. Comparing human factors for augmented reality supported single-user and collaborative repair operations of industrial robots. *Frontiers in Robotics and AI* Volume 6 - 2019. URL: <https://www.frontiersin.org/journals/robotics-and-ai/articles/10.3389/frobt.2019.00037> DOI: 10.3389/frobt.2019.00037

Bern, J. M., May, W. C., Osborn, A., Stella, F., Zargarzadeh, S., Hughes, J., 2024. A Soft Robot Inverse Kinematics for Virtual Reality. In: 2024 IEEE International Conference on Robotics and Automation (ICRA). IEEE, pp. 14957–14963. DOI: 10.1109/ICRA57147.2024.10611603

Borges, E. I. A., Rieder, J. S. I., Aschenbrenner, D., Scharff, R. B. N., 2022. Framework for Armature-Based 3D Shape Reconstruction of Sensorized Soft Robots in eXtended Reality. *Frontiers in Robotics and AI* 9 (April), 1–12. DOI: 10.3389/frobt.2022.810328

Cruz Ulloa, C., Domínguez, D., del Cerro, J., Barrientos, A., 2024. Analysis of MR–VR tele-operation methods for legged-manipulator robots. *Virtual Reality* 28. DOI: 10.1007/s10055-024-01021-z

Eslami, M., Pirmoradian, M., Mokhtarian, A., Baghaei, S., 2023. Design and manufacture of a soft robot with dual-interaction in virtual reality. *Heliyon* 9 (9), e19997. URL: <https://doi.org/10.1016/j.heliyon.2023.e19997> DOI: 10.1016/j.heliyon.2023.e19997

García-Samartín, J. F., Charles, M., del Cerro, J., Barrientos, A., 2024. PETER : a Soft Pneumatic Manipulator with High Resistance and Load Capacity. In: 2024 7th Iberian Robotics Conference (ROBOT). Madrid, pp. 1–6. DOI: 10.1109/ROBOT61475.2024.10796857

García-Samartín, J. F., Martín-Díaz, M., Del Cerro, J., Barrientos, A., 2025. Fast Inverse Kinematics for Soft Manipulators Using TRR Algorithm. In: *Jornadas de Automática*. Cartagena. URL: <https://doi.org/10.17979/ja-cea.2025.46.12253> DOI: <https://doi.org/10.17979/ja-cea.2025.46.12253>

Gong, S., Zudaire, S., Zhang, C., Li, Z., 2025. Augmented reality-enhanced robot teleoperation for collecting user demonstrations. *ArXiv abs/2509.11783*. DOI: 10.48550/arxiv.2509.11783

He, A.-C., Park, J., Beiter, B., Kalita, B., Leonessa, A., 2025. Design and development of a locomotion interface for virtual reality lower-body haptic interaction. *Robotics and Autonomous Systems* 190, 104994. URL: <https://www.sciencedirect.com/science/article/pii/S0921889025000806> DOI: <https://doi.org/10.1016/j.robot.2025.104994>

Makhataeva, Z., Varol, H. A., 2020. Augmented reality for robotics: A review. *Robotics* 9, 21. DOI: 10.3390/robotics9020021

Martín-Barrio, A., Roldán, J. J., Terrile, S., del Cerro, J., Barrientos, A., 2020. Application of immersive technologies and natural language to hyper-redundant robot teleoperation. *Virtual Reality* 24 (3), 541–555. URL: <https://doi.org/10.1007/s10055-019-00414-9> DOI: 10.1007/s10055-019-00414-9

Microsoft, 2019. HoloLens 2 — learn.microsoft.com. <https://learn.microsoft.com/es-es/hololens/hololens2-hardware>. [Accessed 27-02-2026].

Naceri, A., Mazzanti, D., Bimbo, J., Tefera, Y. T., Prattichizzo, D., Caldwell, D., Mattos, L., Deshpande, N., 2021. The vicarious virtual reality interface for remote robotic teleoperation. *Journal of Intelligent & Robotic Systems* 101. DOI: 10.1007/s10846-021-01311-7

Su, Y., Chen, X.-Q., Zhou, C., Pearson, L. R. H., Pretty, C., Chase, J. G., 2023. Integrating virtual, mixed, and augmented reality into remote robotic applications: A brief review of extended reality-enhanced robotic systems for intuitive telemanipulation and telemanufacturing tasks in hazardous conditions. *Applied Sciences*. DOI: 10.3390/app132212129

Wang, X., Guo, S., Xu, Z., Zhang, Z., Sun, Z., Xu, Y., 2024. A robotic tele-operation system enhanced by augmented reality for natural human–robot interaction. *Cyborg and Bionic Systems* 5. DOI: 10.34133/cbsystems.0098