# An overview of XR technologies usage for industrial robot programming

Nikola Živković, Jelena Vidaković, Andrija Dević
*Robotics Department*
*Lola Institute*
Belgrade, Serbia
nikola.zivkovic@li.rs

Mihailo Lazarević
*Department of Mechanics*
*Faculty of Mechanical Engineering*
Belgrade, Serbia
mlazarevic@mas.bg.ac.rs

Ilija Lazarević
*Military Technical Institute*
Belgrade, Serbia
ilija.lazarevic@vti.vs.rs

*Abstract*— **Industrial robot programming can be a challenging task, especially in today's age, where robots are more widespread outside the large manufacturing companies, but rather in small and medium enterprises where users are not necessarily fully qualified individuals. The Extended Reality technologies may be the ongoing answer to improved robot programming experience. Current solutions for robot programming using Extended Reality technologies are explored in this overview. In this paper, a summarized description of certain solutions is given, focusing on how are the XR technologies utilized in developing the robot programming systems. Categorization by devices and motion planners used is also given.**

***Keywords-Virtual reality, augmented reality, extended reality, robot programming***

## I. INTRODUCTION

Virtual Reality (VR), Augmented Reality (AR) and Mixed Reality (MR) are technologies that have rapidly evolved in recent years, revolutionizing the way humans interact with and perceive digital and physical environments. These technologies (collective term Extended Reality - XR) have found applications across various industries such as entertainment, education, healthcare, manufacturing and enterprise [1], [2]. VR headsets (Head Mounted Display - HMD) provide a fully immersive experience in the virtual environment. This technology has proven to be particularly impactful in teleoperated robotics and simulation-based training, where users can experience scenarios that are expensive or pose an increased risk of injury to replicate in the real world. AR overlays digital information onto the real world, enhancing the user's perception of their surroundings. AR applications are commonly experienced through smartphones, tablets, or smart glasses, allowing users to access information or interactive elements in real-time. The Mixed Reality (MR) can be described as a spectrum between VR and AR where user experiences both augmented and virtual realities providing interaction between real-world and virtual objects [3].

Robot programming methods can be classified using different criteria, but the most common classification is based on the user interaction with the robot. This classification

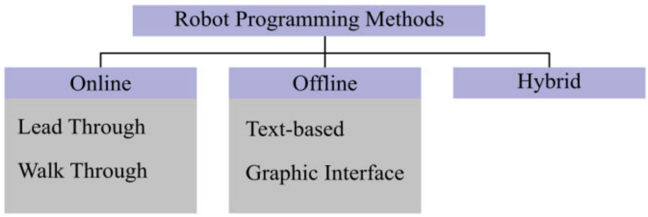


Figure 1. Robot programming methods summary. Adapted from [5].

recognizes online, offline, and hybrid robot programming [4], [5] (see Figure 1.). Online programming methods require a user to interact with a real robot during programming, while offline methods don't have that requirement. Hybrid methods are a combination of online and offline programming. The online programming methods group consists of *lead-through* and *walk-through* methods [6]. Offline programming is usually text-based or uses a graphical interface. The aforementioned methods are well established, especially in the manufacturing industry; however, they are time-consuming and often require expert knowledge. During the past two decades, progress has been made in developing robot programming techniques using XR technologies. XR technologies have major potential to advance current and enable the development of new robot programming methods. The XR devices became more available to a wider range of customers every year, enabling recent increased research and development in robotics and manufacturing.

This paper reviewed current XR-based solutions for robot programming focusing on industrial robotics. The paper is divided into three sections, with the first section being the introduction; the second section contains the majority of the review, which is divided further into subsections based on the type of XR technology. The third section is the conclusion.

## II. XR SOLUTIONS FOR ROBOT PROGRAMMING

Considerable variability exists in the conceptualization of XR-based robot programming techniques, and the following subsections will present some of the recent solutions.

## A. *VR robot programming solutions*

Considering that VR technology imposes an immersive environment experience on a user, one of its main use cases in the industry is personnel training [7], [8].

VR is also used in remote robot programming [9] and teleoperation-based control [10]. The design presented in [9] focuses on easy and time-efficient robot programming and repurposing without the operator's presence in the robot workspace. The operator is fully immersed in the VR environment using a robocentric perspective, meaning that the operator has the observer's point of view. The interface for robot programming is divided into blocks representing the robot's actions. The robot is programmed by combining these blocks into sequences. The complete sequence is validated in simulation mode and sent to the real robot. The VR solution [10] is teleoperation-based robot control with assistance provided to the user in certain situations. The assistance part is conceptually the same as the robot programming using a sequence of defined waypoints. If the sequence is valid, it is executed by the robot, and upon finishing, the control over the robot is released back to the human operator (user). The article [11] presents robot programming where the main premise is that the robot recreates the desired motion generated by a human user doing some tasks in the VR environment interacting with virtual objects. The case study presented in [11] is a turbine part cleaning task, where the human user in a VR environment replicates the cleaning motion on the virtual turbine part and motion is recorded. The recorded motion is used as a desired trajectory for the robot when a real task is executed.

## B. *AR/MR robot programming solutions*

The full virtual environment of the VR technology isn't suitable for on-site usage. The virtual augmentation of the real world presented to users through an Augmented and Mixed Reality (AR/MR) Head-Mounted Display (HMD) or any other display device is a better solution for robot programming tools in a factory setting. Table I. summarizes the AR/MR solutions based on the XR interface. Many AR/MR applications for robot programming use the concept of programming by demonstration [12], enabling users to program robots with minimal knowledge, and compared to VR solutions, there is no need to model the entire workshop.

TABLE I. ROBOT PROGRAMMING SOLUTIONS CATEGORIZED BY XR INTERFACE

| XR interface | Reference |
|---|---|
| HMD | [9], [13-26] |
| Monitor/Tablet/ Smartphone | [27-32] |

The MR solution in [13] uses the *walk-through* programming method by defining a desired robot's end-effector path as a set of waypoints. Each waypoint describes the position and orientation of the end-effector and can be translated and/or rotated for editing purposes. Additionally, for waypoints, the user has the option to define gripper actions such as opening and closing. The motion between waypoints can be defined as a PTP (Point-To-Point) or CP (Continuous Path) motion. Concerning MR features, the solution [13] offers the walk-through mode by drawing path instead of definition by waypoints. The authors of a similar MR solution in [14] investigated the efficiency and intuitiveness of MR-based robot programming compared to programming using a 2D interface. The authors of this research designed tests for users in both interfaces. The MR-based programming interface turned out to be easier to use, and completing designated test tasks was faster than using the 2D interface.

AR application in [15] focuses on a high-level task definition by a user. The authors of [15] presented an AR-based robot programming application where the robot is programmed by a user manipulation of virtual objects, which are the representation of the real-world objects. The real-world objects are intended to be subject to the real robot pick-and-place operations. Namely, the user defines tasks by moving virtual objects, utilizing hand gestures which AR device recognizes. Moving the virtual object from one place to another defines the starting and goal positions and orientations for the robot end-effector inside the robot workspace. The motion planning module checks the task's validity and generates solution for the user to preview. Although programming is easier, increased flexibility in manufacturing tasks can be detrimental. The AR programming system in [16] is developed with a focus on programming the robot to carry out assembly tasks. The parts to be assembled are designated with markers so that the AR device can recognize them. The AR device displays the virtual parts to the user to overlay the real ones. The desired trajectory for the robot is programmed by a user who assembles the virtual parts through the pick-and-place process using an Air-Tap gesture to pick and move objects. Based on the virtual assembly, the start and end coordinates of the end effector are sent to the robot motion planner, and the desired trajectory is generated via the KukaRobotLanguage since the Kuka KR6 R900 robot is used in this study. The AR solution in [17] uses HoloLens' Air-Tap gesture to drag and drop the virtual end-effector to the desired pose. The motion planning module checks the validity of the planned trajectory and, based on the outcome, either resets the planning process or executes the motion.

There are different approaches to how Tool Centre Point (TCP) is controlled by a user in AR/MR applications. Some applications use Air-Tap and Drag-and-Drop gestures inside the AR environment to position the TCP [15-17], [19-24], [26], while others use handheld pointers, a real representation of the virtual TCP [18]. The solutions presented in [18] use a handheld pointer to represent the position and orientation of the robot end-effector in a virtual environment. The user programs the robot by placing the pointer at desired points thus defining the position and orientation of the robot end-effector at that point in the robot workspace. The system can also create CP motion by clicking the pointer's button and moving the pointer through space. Every movement of the pointer is displayed through the AR interface as a movement of the virtual robot end effector along with trajectory planning data. The noise in the trajectory caused by a human hand tremor is filtered with smoothing filters. The AR-assisted system in [24] utilizes hand gestures for robot programming. The interface relies solely on AR device built-in hand gestures without using any buttons. The desired trajectory is drawn by a user using the Air-Tap gesture. The indication that trajectory is being successfully recorded is presented as a green sphere at the tip of the index finger, while a red sphere indicates that the system doesn't record. The custom-developed ROS-based translators translate the recorded desired trajectories into specific robot programming language with given examples for UR5 and ABB IRB 2600 robots.

Smartphone-based AR applications for robot programming [27], [28], [33] are cost-effective solutions compared to HMD that can be used in small and medium enterprises (SMEs) and research. In smartphone-based AR applications, reduced cost is traded for reduced visual representation and speed of programming. The solution described in [33] presents simple Android-based application intended to be used as a verification tool after the trajectory is planned in RoboDK [34]. Android app with the developed GUI installed on smartphone can be also used to simulate teaching pendant for programming by the *lead-through* method (see Figure 2.).

Motion planning is an important part of robot programming system. While XR-assisted solutions enable users to program robots with as little expertise as possible, it usually means that the motion planner has to ensure smooth and collision-free motion of the robot between points designated by a user. The majority of the reviewed solutions use the open-source motion planner – MoveIt. Table II shows other motion planners used besides MoveIt.

TABLE II. ROBOT PROGRAMMING SOLUTIONS CATEGORIZED BY MOTION PLANNER

| Motion planner | Reference |
|---|---|
| ROS-MoveIt | [9], [13-15],[17-19] |
| RoboDK | [33] |
| Custom | [31] |
| KukaRobotLanguage | [16] |

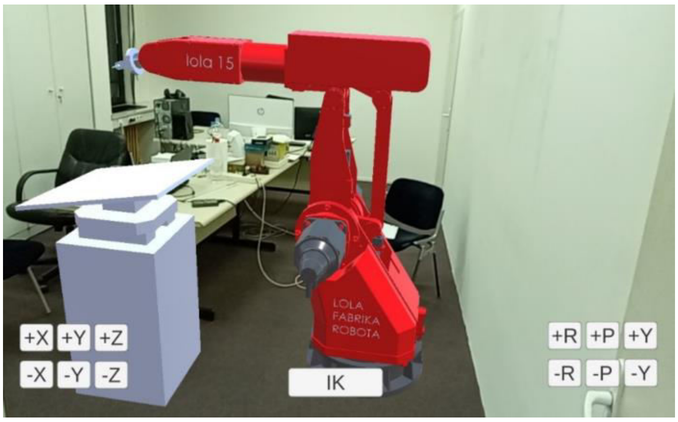


Figure 2. GUI of AR-based Android robot programming application. Adapted from [33].

## III. CONCLUSION

The XR technologies possess great potential to be implemented as a robot programming tools in manufacturing and other service industries that rely on robotic automation. The programming ease and intuitiveness of a reviewed solutions promises great future especially for small and medium enterprises where operator training cost can be significant.

## ACKNOWLEDGMENT

This research has been supported by the research grants of the Serbian Ministry of Science, Technological Development and Innovations, grant No. 451-03-65/2024-03/200105 from 5.2.2024 and No. 451-03-66/2024-03/200066.

## REFERENCES

[1] D. C. Villagran-Vizcarra, D. Luviano-Cruz, L. A. Pérez-Domínguez, L. C. Méndez-González, and F. Garcia-Luna, “Applications Analyses, Challenges and Development of Augmented Reality in Education, Industry, Marketing, Medicine, and Entertainment,” *Applied Sciences*, vol. 13, no. 5, Art. no. 5, Jan. 2023, doi: 10.3390/app13052766.

[2] M. Soliman, A. Pesyridis, D. Dalaymani-Zad, M. Gronfula, and M. Kourmpetis, “The Application of Virtual Reality in Engineering Education,” *Applied Sciences*, vol. 11, no. 6, Art. no. 6, Jan. 2021, doi: 10.3390/app11062879.

[3] S. Rokhsaritalemi, A. Sadeghi-Niaraki, and S.-M. Choi, “A Review on Mixed Reality: Current Trends, Challenges and Prospects,” *Applied Sciences*, vol. 10, no. 2, Art. no. 2, Jan. 2020, doi: 10.3390/app10020636.

[4] B. Siciliano and O. Khatib, Eds., *Springer Handbook of Robotics*. in Springer Handbooks. Cham: Springer International Publishing, 2016. doi: 10.1007/978-3-319-32552-1.

[5] O. Heimann and J. Guhl, “Industrial Robot Programming Methods: A Scoping Review,” in *2020 25th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, Sep. 2020, pp. 696–703. doi: 10.1109/ETFA46521.2020.9211997.

[6] A. Gaschler, M. Springer, M. Rickert, and A. Knoll, “Intuitive robot tasks with augmented reality and virtual obstacles,” in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, May 2014, pp. 6026–6031. doi: 10.1109/ICRA.2014.6907747.

[7] P. Karagiannis, T. Togias, G. Michalos, and S. Makris, “Operators Training Using Simulation And VR Technology,” *Procedia CIRP*, vol. 96, pp. 290–294, Jan. 2021, doi: 10.1016/j.procir.2021.01.089.

[8] D. M. Oliveira, S. C. Cao, X. F. Hermida, and F. M. Rodriguez, “Virtual Reality System for Industrial Training,” in *2007 IEEE International Symposium on Industrial Electronics*, Jun. 2007, pp. 1715–1720. doi: 10.1109/ISIE.2007.4374863.

[9] T. Togias, C. Gkournelos, P. Angelakis, G. Michalos, and S. Makris, “Virtual reality environment for industrial robot control and path design,” *Procedia CIRP*, vol. 100, pp. 133–138, 2021, doi: 10.1016/j.procir.2021.05.021.

[10] C. Barentine, A. McNay, R. Pfaffenbichler, A. Smith, E. Rosen, and E. Phillips, “A VR Teleoperation Suite with Manipulation Assist,” in *Companion of the 2021 ACM/IEEE International Conference on Human-Robot Interaction*, in HRI ’21 Companion. New York, NY, USA: Association for Computing Machinery, Mar. 2021, pp. 442–446. doi: 10.1145/3434074.3447210.

[11] A. Burghardt, D. Szybicki, P. Gierlak, K. Kurc, P. Pietruś, and R. Cygan, “Programming of Industrial Robots Using Virtual Reality and Digital Twins,” *Applied Sciences*, vol. 10, no. 2, Art. no. 2, Jan. 2020, doi: 10.3390/app10020486.

[12] G. Biggs and B. MacDonald, “A Survey of Robot Programming Systems,” in *Proceedings of the Australasian conference on robotics and automation*, Dec. 2003, pp. 1–3.

[13] M. Ostanin, S. Mikhel, A. Evlampiev, V. Skvortsova, and A. Klimchik, “Human-robot interaction for robotic manipulator programming in Mixed Reality,” in *2020 IEEE International Conference on Robotics*

*and Automation (ICRA)*, May 2020, pp. 2805–2811. doi: 10.1109/ICRA40945.2020.9196965.

[14] S. Y. Gadre, E. Rosen, G. Chien, E. Phillips, S. Tellex, and G. Konidaris, "End-User Robot Programming Using Mixed Reality," in *2019 International Conference on Robotics and Automation (ICRA)*, May 2019, pp. 2707–2713. doi: 10.1109/ICRA.2019.8793988.

[15] J. D. Hernandez, S. Sobti, A. Sciola, M. Moll, and L. E. Kavraki, "Increasing Robot Autonomy via Motion Planning and an Augmented Reality Interface," *IEEE Robot. Autom. Lett.*, vol. 5, no. 2, pp. 1017–1023, Apr. 2020, doi: 10.1109/LRA.2020.2967280.

[16] S. Blankemeyer, R. Wiemann, L. Posniak, C. Pregizer, and A. Raatz, "Intuitive Robot Programming Using Augmented Reality," *Procedia CIRP*, vol. 76, pp. 155–160, Jan. 2018, doi: 10.1016/j.procir.2018.02.028.

[17] K. Lotsaris, C. Gkournelos, N. Fousekis, N. Kousi, and S. Makris, "AR based robot programming using teaching by demonstration techniques," *Procedia CIRP*, vol. 97, pp. 459–463, Jan. 2021, doi: 10.1016/j.procir.2020.09.186.

[18] S. K. Ong, A. W. W. Yew, N. K. Thanigaivel, and A. Y. C. Nee, "Augmented reality-assisted robot programming system for industrial applications," *Robotics and Computer-Integrated Manufacturing*, vol. 61, p. 101820, Feb. 2020, doi: 10.1016/j.rcim.2019.101820.

[19] A. Rivera-Pinto, J. Kildal, and E. Lazkano, "Toward Programming a Collaborative Robot by Interacting with Its Digital Twin in a Mixed Reality Environment," *International Journal of Human–Computer Interaction*, pp. 1–13, Jun. 2023, doi: 10.1080/10447318.2023.2221599.

[20] D. Araiza-Illan, A. De San Bernabe, F. Hongchao, and L. Y. Shin, "Augmented Reality for Quick and Intuitive Robotic Packing Re-Programming," in *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, Mar. 2019, pp. 664–664. doi: 10.1109/HRI.2019.8673327.

[21] D. Bambusek, Z. Materna, M. Kapinus, V. Beran, and P. Smrz, "Combining Interactive Spatial Augmented Reality with Head-Mounted Display for End-User Collaborative Robot Programming," in *2019 28th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, New Delhi, India: IEEE, Oct. 2019, pp. 1–8. doi: 10.1109/RO-MAN46459.2019.8956315.

[22] J. Guhl, S. Tung, and J. Kruger, "Concept and architecture for programming industrial robots using augmented reality with mobile devices like microsoft HoloLens," in *2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, Sep. 2017, pp. 1–4. doi: 10.1109/ETFA.2017.8247749.

[23] M.-E. Matour and A. Winkler, "Intuitive Robot Path Planning through Augmented Reality," in *2023 27th International Conference on Methods and Models in Automation and Robotics (MMAR)*, Aug. 2023, pp. 27–32. doi: 10.1109/MMAR58394.2023.10242522.

[24] I. Soares, M. Petry, and A. P. Moreira, "Programming Robots by Demonstration Using Augmented Reality," *Sensors*, vol. 21, no. 17, p. 5976, Sep. 2021, doi: 10.3390/s21175976.

[25] Y. Sarai and Y. Maeda, "Robot programming for manipulators through volume sweeping and augmented reality," in *2017 13th IEEE Conference on Automation Science and Engineering (CASE)*, Aug. 2017, pp. 302–307. doi: 10.1109/COASE.2017.8256120.

[26] C. P. Quintero, S. Li, C. Shing, W. Chan, and S. Sheikholeslami, "Robot Programming Through Augmented Trajectories," 2018.

[27] C. Mateo, A. Brunete, E. Gambao, and M. Hernando, "Hammer: An Android based application for end-user industrial robot programming," in *2014 IEEE/ASME 10th International Conference on Mechatronic and Embedded Systems and Applications (MESA)*, Sep. 2014, pp. 1–6. doi: 10.1109/MESA.2014.6935597.

[28] S. M. Chacko, A. Granado, and V. Kapila, "An Augmented Reality Framework for Robotic Tool-path Teaching," *Procedia CIRP*, vol. 93, pp. 1218–1223, 2020, doi: 10.1016/j.procir.2020.03.143.

[29] H. C. Fang, S. K. Ong, and A. Y. C. Nee, "Novel AR-based interface for human-robot interaction and visualization," *Adv. Manuf.*, vol. 2, no. 4, pp. 275–288, Dec. 2014, doi: 10.1007/s40436-014-0087-9.

[30] S. K. Ong, A. Y. C. Nee, A. W. W. Yew, and N. K. Thanigaivel, "AR-assisted robot welding programming," *Adv. Manuf.*, vol. 8, no. 1, pp. 40–48, Mar. 2020, doi: 10.1007/s40436-019-00283-0.

[31] M. Kapinus, Z. Materna, D. Bambušek, V. Beran, and P. Smrž, "ARCOR2: Framework for Collaborative End-User Management of Industrial Robotic Workplaces using Augmented Reality." arXiv, Jun. 14, 2023. Accessed: Nov. 06, 2023. [Online]. Available: http://arxiv.org/abs/2306.08464

[32] D. Ni, A. W. W. Yew, S. K. Ong, and A. Y. C. Nee, "Haptic and visual augmented reality interface for programming welding robots," *Adv. Manuf.*, vol. 5, no. 3, pp. 191–198, Sep. 2017, doi: 10.1007/s40436-017-0184-7.

[33] J. Vidaković, A. Dević, I. Lazarević, and N. Živković, "Design of augmented reality-based Android app for simulation and programming of industrial robots," in *Proceedings of the International Conference of Experimental and Numerical Investigations and New Technologies, CNNTech 2023*, Zlatibor, Serbia, Jul. 2023, vol. 792.

[34] "Simulator for industrial robots and offline programming - RoboDK." Accessed: Dec. 18, 2023. [Online]. Available: https://robodk.com/