Intuitive Augmented Reality Interface for Non-Expert Robot Programming

Joosun Lee, Maram Sakr, Taeyhang Lim, H.F. Machiel Van der Loos, Wansoo Kim

***Abstract*—This study proposes an intuitive augmented reality (AR) interface for robot programming, targeting non-expert users. Traditionally, AR interfaces were employed for simple robot control tasks, with interaction techniques primarily explored within these limited contexts. Previous researchers have concluded that using multi-modal interaction methods that combine two or more interaction techniques is more effective than a single modality. However, these studies had the limitation of only testing simple tasks. In response to this limitation, our paper introduces an AR interface designed for the complex task of controlling both the trajectory and null space of a robot. In this study, users were tasked with performing a peg-in-hole task while avoiding obstacles. Two selection methods were compared: pointing and gazing. The pointing method, considered single-modal, utilized only hand gestures, while the gazing method was multi-modal, incorporating both gaze and hand movements. The results indicated improved performance with the pointing method, although the difference was not statistically significant. Interestingly, users significantly preferred the pointing method. This suggests that, for non-expert users handling complex robot tasks, the pointing method offers a more intuitive control mechanism compared to gazing.**

***Index Terms*—Human-Robot Interaction, Augmented Reality, Null Space Control**

I. INTRODUCTION

The collaboration between humans and robots is undergoing significant evolution. With the increasing presence of robots in everyday scenarios, ranging from restaurant service to doorstep package deliveries, their integration into diverse domains is becoming increasingly evident [1]. Nevertheless, despite their growing prevalence, achieving optimal human-robot collaboration remains a challenge. This challenge involves creating natural and intuitive user interfaces that are easy for non-experts to learn and use [2]. Thus, in this paper, we aim to develop natural and intuitive user interfaces for non-experts to use (Fig. 1).

Traditionally, robots were operated through programming only. This meant that if users were non-experts, they could only run pre-set programs created by experts, resulting in highly repetitive and less interactive human-robot collaboration. Consequently, the focus shifted to methods such as using joysticks and teach pendants for non-experts to control the robot [3], [4]. Additionally, augmented reality (AR) has proven helpful in allowing non-experts to interact with robots more efficiently and intuitively. AR serves as an emerging, user-friendly technology that facilitates human interaction with virtual holograms in real-world settings [5], notably enhancing human-robot collaboration in robotics [6], [7].

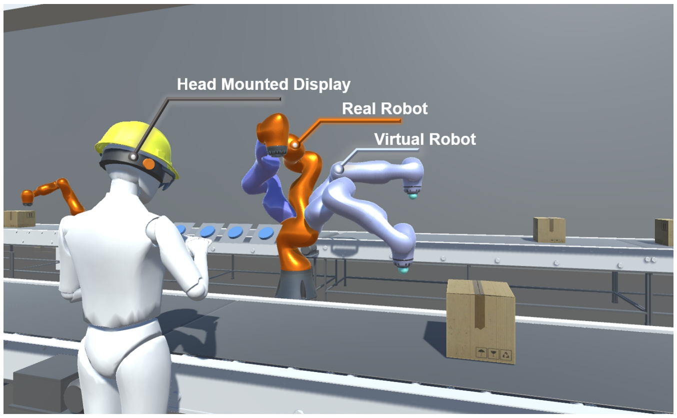

Fig. 1. A non-expert user utilizes HoloLens for robot task and motion planning in an industrial setting, employing waypoint navigation and position control. The augmented reality interface projects potential robot pathways, while null space control allows for configuration adjustments without affecting the end-effector position, ensuring safe and efficient task execution.

For instance, an AR keyboard allowed the classification of an object's name for later use in training data [8], enabling the robot to identify the object later on by name. Furthermore, researchers have worked to facilitate natural and intuitive communications for humans to enhance user experience. For this, studies have used modality cues, including gestures, voice, and gaze, for robot control [9]. In a related study [10], compared pinch, button clicks, and dwell time as selection methods for virtual objects in combination with an eye-gaze system. They concluded that pinch gestures presented a reasonable alternative to button clicks. This demonstrated the effectiveness of the pinch gesture in selecting virtual objects. Additionally, in many studies, these communication techniques are often combined to allow more natural interactions [11]–[13]. Notably, multi-modal cues—combining two or more modalities—have demonstrated higher efficacy compared to singular cues [14], [15].

However, challenges still exist with the current robot control system through AR. Existing research tends to focus on singular functions rather than multiple functions [16], [17]. This means that the AR robot controller involves a simple task (i.e. pick and place) rather than more complex tasks of a combination of different tasks, like avoiding obstacles while performing pick and place tasks. From this, the modality interaction is still questionable as well when the system

---

This paper was produced by the IEEE Publication Technology Group. They are in Piscataway, NJ.

Manuscript received April 19, 2021; revised August 16, 2021.

becomes more complex. Thus, there is a need for a more complex system that could be operated on a robot using AR as well as a need to evaluate the interaction cues for a complex task.

In summary, the main contributions of the paper are:

1) A user-friendly AR-based interface for robot programming featuring four interactive functions: 1) position control, 2) virtual preview of robot motion, 3) trajectory modification, and 4) null space control.
2) A comparative study of two modalities for object selection in an AR environment: pointing and gazing.

II. RELATED WORK

The key to human-robot collaboration (HRC) is maintaining safety and productivity, and for this, communication between the robot and the human is crucial. AR technology enables intuitive control of digital systems and finds applications across diverse fields, including education [18], medicine [19], and industry [20], notably enhancing human-robot collaboration in robotics [7]. The efficiency of AR in robotics has been demonstrated through a comparison between AR and a joystick for manipulating a robot [11]. In this comparison, AR users wear a head-mounted device (HMD) called the HoloLens to interact with holograms to control the robot. The conclusion was that AR reduced physical demand and performance time, making the robot more convenient to use.

Various research studies have compared different interaction cues to increase naturalness and intuitiveness for users. For example, studies have compared single-mode to multi-modal techniques, where, [13] compared four combinations of gaze and hand: Gaze&Finger, Gaze&Hand, Gaze&Pinch, and Point&Pinch. Users were instructed to select the target block on the menu displayed as holograms under each condition. Results indicated that gaze integration showed higher performance compared to the hand-only (Point&Pinch) technique. Another study compared gaze-only and hand-only techniques with combined gaze-hand techniques, demonstrating that users improved both performance and user experience [12]. While these studies highlight the effectiveness of multi-modal approaches using AR, the tasks evaluated were relatively simple, involving button selection and pick-and-place. Therefore, the evaluation of interaction techniques for complex tasks involving several steps remains an open question.

Moreover, the lack of complex robot control systems persists as existing research tends to focus on singular abilities rather than multiple functions. For example, in [16], it was shown how users could create robot trajectories and preview robot motions using an AR interface. Although position control is crucial, relying solely on this function increases collision risks. To address this, null space control becomes essential, allowing the robot to maintain its end-effector location with a new joint configuration [21]. Integrating null space control with AR, as described in [17], allows the robot to reach a desired point for the end-effector and control each joint to avoid collisions. However, this study also focuses on a single function of null space and does not combine it with other functions. Thus, a more complex system combining position control, trajectory generation, and null space control is still in demand in research.

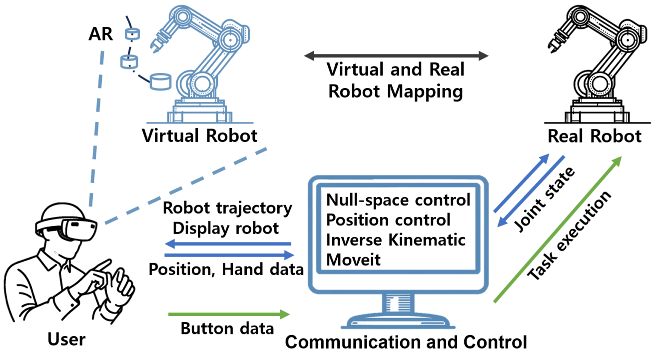

Fig. 2. Framework of the components in the proposed method.

In summary, this study aims to explore two different interaction techniques—pointing and gazing—with the purpose of experimenting on whether non-expert users can efficiently operate this complex system of combined gestures.

III. SYSTEM

The purpose of this study is to enable the control of an industrial robot through augmented reality (AR). Our work aims to address the limitations of simplicity in AR robot control by incorporating a combination of different control methods rather than relying on just one. Therefore, it encompasses both position and null space control of the manipulator. This study uses the KUKA LBR iiwa 14 manipulator that has 7 degrees of freedom (DOF).

For AR interactions, we utilize the Microsoft Hololens 2 [22] for both interaction techniques. The overall framework of this study is illustrated in Fig. 2. The user, wearing the HoloLens 2, positions waypoints using holograms. Upon pressing the ”simulate” button, the virtual robot follows the path created through these waypoints. The execution of the robot’s movements is handled through ROS (Robot Operating System) after computation. Subsequently, obstacles are introduced along the pathway. The user adjusts the waypoints (position control) and joint positions (null space control) to facilitate obstacle avoidance. These modifications are updated in ROS, and once the user is satisfied with the edited joint states, they can proceed to execute the commands on the real robot.

A. Augmented Reality

Augmented reality (AR) is developed using the Unity 3D game engine, which can be programmed using C# scripts. The AR scene is deployed on the Microsoft HoloLens 2 to enable user interaction. Through the Mixed Reality Toolkit (MRTK), hand and gaze tracking is made possible, allowing users to make selections, move holograms, and recognize pinch gestures.

Upon wearing the HoloLens 2 for the first time, users can visualize the mapped AR robot overlaid on the real robot. Synchronization between the AR robot and the physical robot is achieved using a QR code provided by Microsoft [23].

Users can interact with the hologram buttons displayed on the HoloLens 2 for robot controls. The system features three sets of buttons: (1) Edit and execute: edit waypoint, edit joint, simulation, real robot; (2) Edit waypoint: add waypoint, delete waypoint, reset waypoint, save; (3) Edit joint: save.

The waypoints represent the shape of the end-effector of the robot. For enhanced visual information of the holograms, the colors of the buttons, waypoints, and the robot in the AR scene change when hovered over or selected. The functions of each button are explained thoroughly throughout this section.

B. Robot Operating System

To operate both the AR and the real robot, information regarding waypoints and commands is communicated from the HoloLens to the Robot Operating System (ROS). This communication is facilitated through the ROS# library. For the robotic tasks within ROS, the Moveit library is employed. Moveit plays a crucial role by planning the motion of the robot arm, considering its degrees of freedom and environmental factors. It allows the robot to precisely control its desired position and visualize its movements within a simulation environment. This capability enables the robot to perform tasks along a pre-planned path. Moveit is versatile and can be applied to various robot platforms, facilitating efficient robot control and task automation.

When the user presses a button, essential information is exchanged between the HoloLens and ROS. In edit waypoint mode, pressing the ”save” button communicates the waypoint positions to ROS. In edit and execute mode, pressing the ”simulation” button initiates the execution of the virtual robot while pressing the ”real robot” button executes the real robot. The ”edit joint” button receives joint states from ROS to display the robot in the corresponding states at each waypoint position. When the user edits the joint and presses the ”save” button, the edited joint states are transmitted to ROS and are utilized in the subsequent robot execution.

C. Trajectory Planning in Cartesian Space

Before trajectory planning, the robot's end-effector pose is calculated through equation 1 using the joint angles θ_i to find the robot's initial position. The matrix $\boldsymbol{M}$ is comprised of a position vector $\mathbf{t}$ in $\mathbb{R}^3$ and a rotation matrix $\boldsymbol{R}$ within SO(3). These components jointly describe the pose of the end-effector relative to the base coordinate system.

$$\prod_{i=1}^{7} M_i(\theta_i). \tag{1}$$

To achieve Cartesian trajectory planning, a critical step involves converting the planned path in Cartesian space into joint space, using the process of inverse kinematics. This conversion allows determining the path each joint must follow to achieve the desired end-effector pose.

The practical implementation of Cartesian space trajectory planning is facilitated through the Moveit *move group interface* (C++) API and KDL (Kinematics and Dynamics Library). By defining waypoints along the desired path, it becomes possible to plan the robot's reachability and subsequently execute motion planning and control for the physical robot arm.

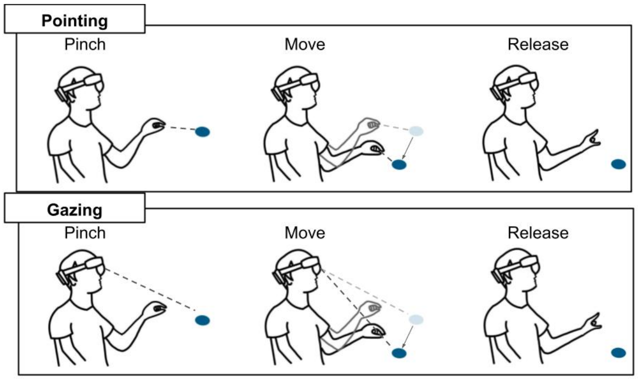

Fig. 3. Interaction Techniques used in this study. Top: user selects the hologram through pointing and pinching using their hands. Bottom: user gazes at the hologram of interest and pinches their hands to select it.

Moveit also provides a valuable feature called *display planned path*, enhancing the visualization of the robot's control and motion planning processes. This visualization is a critical tool for validating the alignment of the robot's operational path with the intended trajectory. Additionally, it aids in identifying and addressing potential obstacles, contributing to effective collision prevention during the motion planning phase.

For trajectory planning using AR, users first create waypoints for the pathway. Pressing the ”edit waypoint” button allows users to add waypoints through the ”add waypoint” button. Also, using the ”delete waypoint” and ”reset waypoint” buttons, users can delete the latest waypoint or delete all waypoints, respectively. To select waypoints, users use a pinch gesture and move objects by moving their hand while pinched. When the waypoint is at the desired location, users release their pinched fingers. Pressing the ”save” button transitions the set of buttons to the ”edit and execute” mode.

D. Null space Control

The null space control of the robot is defined as using the redundancy of the robot to allow different joint positions while keeping the end-effector location stationary. The differential kinematics equation establishes a linear relationship between joint space velocities and task space velocities. This relationship can be employed to calculate joint velocities using the kinematic equation. Consequently, the differential kinematics equation is used to determine valid joint trajectories based on position and velocity information. The equation is represented as follows:

$$\dot{\boldsymbol{x}} = \boldsymbol{J}(\boldsymbol{q})\dot{\boldsymbol{q}} \tag{2}$$

Because of the non-square Jacobian matrix in a 7-DOF manipulator, the fundamental inverse solution to equation 2 is derived by utilizing the pseudoinverse $\boldsymbol{J}^\dagger$ of the matrix $\boldsymbol{J}$. Consequently, the inverse solution can be expressed as follows:

$$\dot{\boldsymbol{q}} = \boldsymbol{J}^\dagger(\boldsymbol{q})\dot{\boldsymbol{x}} \tag{3}$$

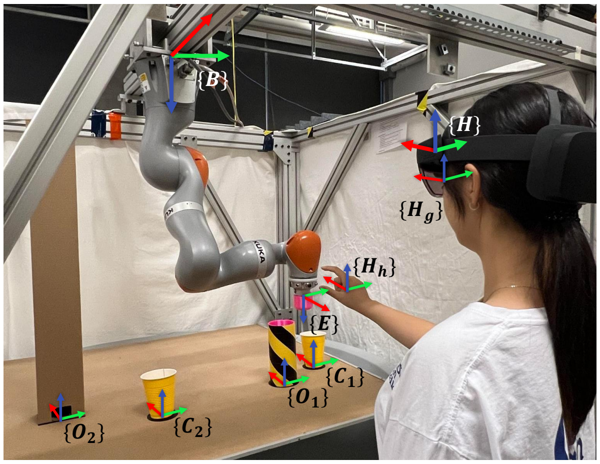

Fig. 4. The experiment setup for the robot control using the HoloLens 2, H. The setup consists of 2 obstacles O, 2 peg-in-hole cups C and the robot (base B and end-effector E). The user uses their hands H_h and gaze H_g to interact with the robot.

The core equation governing null space control is given by [24]:

$$\dot{\boldsymbol{q}} = \boldsymbol{J}^{\dagger}(\boldsymbol{q})\dot{\boldsymbol{x}} + (\boldsymbol{I} - \boldsymbol{J}^{\dagger}(\boldsymbol{q})\boldsymbol{J}(\boldsymbol{q}))\dot{\boldsymbol{q}}_{\boldsymbol{g}} \tag{4}$$

To engage in the null space control of the robot, the user can press the ”edit joint” button. After pressing the button, for each waypoint created, a robot image will be shown in the state of the waypoint (i.e., when 10 waypoints are created, there will be 10 robots). The user selects the robot of interest by selecting the corresponding waypoint. When the waypoint is selected, the robot moves to that position. Users can then pinch the corresponding robot and slide their hand left or right. The displacement of the pinched hand is calculated to be delivered to the robot. The displacement is integrated into $\dot{q_g}$ in equation 4 and it adjusts the joint velocity of the robot while keeping its task space stationary.

E. Interaction Techniques

Two interaction techniques are used for comparison: pointing and gazing. The difference between the two interaction techniques is that for pointing, it is a single-modal method, requiring only the user’s hands. For this method, a white dashed line extends from the user’s palm to hover over the hologram of interest, and users can pinch to make the selection. In contrast, gazing is a multi-modal method where users can gaze at the hologram of interest and make the selection using their pinches. In both methods, the button or object of the hovered hologram changes color for visual information to the user, and the holograms can be moved by the user’s hand while pinching the object of interest.

IV. EXPERIMENT

In this experiment, the KUKA LBR iiwa (Intelligent Industrial Work Assistant) 14, featuring 7 degrees-of-freedom, was utilized. This robot, being a redundant manipulator, enables null space control. It was positioned in an inverted orientation on a stationary platform, approximately 1.8 m x 1.7 m x 1.9 m in size. The robot platform remained static throughout the experiment, with only the joints being actively controlled.

A. Participants

We recruited participants for the user study through word-of-mouth, advertisements posted on the university campus, and social media. A total of 16 participants (11 males, 4 females, and 1 preferring not to disclose their gender), with an average age of 24.6, participated in our study. Before conducting the study, we obtained research ethics approval from the University of British Columbia Behavioural Research Ethics Board (application ID H20-03740). We obtained informed consent from each participant before commencing each experiment session.

All participants had normal or corrected-to-normal vision. The order of experiencing the interaction cues was counterbalanced among participants to mitigate carryover effects. Before the experiment, instructions on how to operate the system and a short demonstration video[1] were provided to the users.

B. Experiment Task

The experiment consisted of two peg-in-hole tasks of inserting a peg attached to the robot’s end-effector into a cup and then removing it from the cup. The peg used in the experiment was cylindrical, with a diameter of 7 cm and a height of 5 cm. The cups used in the peg-in-hole tasks had dimensions of 6 cm bottom diameter, 12 cm height, and 9 cm top diameter. After, the pathway was disrupted using two obstacles and the users were asked to adjust the robot’s trajectory and the joint space to avoid the obstacles. Obstacle 1, which disrupted the task space of the robot, was cylindrical (9 cm x 20 cm), while Obstacle 2, which interfered with the robot’s joint space, was rectangular (11 cm x 8 cm x 90 cm). The location of the cups and the obstacles were consistent among participants.

C. Procedure

First, the users were to engage in the ”tips” provided on the HoloLens 2 to learn how to operate the device as well as to calibrate their eye gaze. After that, the users were asked to perform two peg-in-a-hole tasks by creating a pathway using the AR waypoints. The waypoints and other joint controls were adjusted through the holograms including the buttons displayed on the HoloLens 2. The order in which the waypoints were created was the order the robot moved.

To create a complete path and perform the peg-in-hole task, users were instructed to make to make a certain number of waypoints: three waypoints for each cup (above the cup, inside the cup and then outside of the cup again), and then a minimum of three waypoints between the two cups.

After creating a path through the waypoints (1), users can simulate the AR robot to verify whether it follows the correct path and successfully completes the peg-in-hole task (2). Subsequently, once the user is satisfied with the path, two

---

[1] https://www.youtube.com/shorts/bVDakXiW0D8

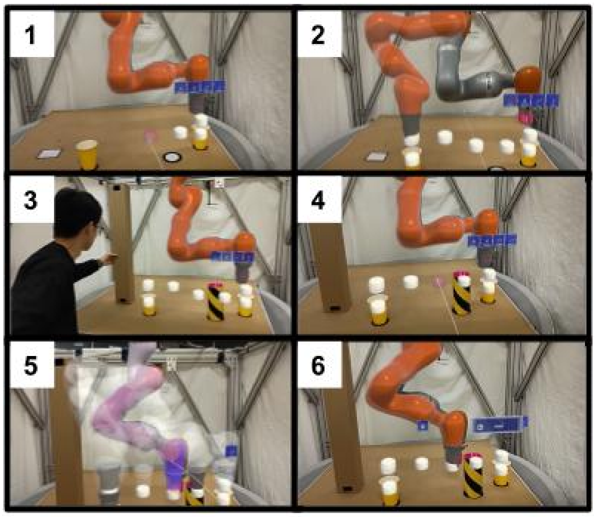

Fig. 5. An example of the experimental procedure using the pointing method. The user first creates a pathway for the peg-in-hole task (1) and virtually visualizes the simulated robot (2). After obstacle insertion (3), the user adjusts both the pathway (4) and the null space (5) to avoid the obstacles. Afterward, the task is executed with the real robot (6).

obstacles will be introduced in the path (3). Participants will then need to modify the path (4) and adjust the null space (5) to avoid any collisions. After these edits, users will execute the task with the real robot (6). This procedure is illustrated in Figure 5.

After the robot control experiment, the users participated in a questionnaire for each condition. Once they completed the questionnaires, they proceeded to the next experiment, which involved a different interaction technique. After completing both conditions and the questionnaires, the users were also asked to give general feedback on this study.

V. ANALYSIS

When evaluating new user interfaces, it is essential to examine the system from both objective and subjective perspectives. Objective measures allow us to determine if the new interface brings measurable performance improvements to the process. On the other hand, subjective measures help us understand users' perceptions of the interface's use, such as perceived task load and usability. Subjective measures are crucial, as the users' perception of the interface is closely related to their likelihood of accepting the technology. This acceptance is key in determining whether the technology will continue to be utilized and further developed [25], [26]. We use the following objective and subjective measures to provide a multi-dimensional evaluation of the proposed interface with different modalities.

A. Objective Measures

1) **Task Completion Time:** We measured the total time in minutes required by the user to complete the experiment task under each condition.
2) **Task Success:** This metric calculates the frequency of the user successfully programming the robot to insert the peg into the correct hole for each condition. It is normalized by the total number of holes to bring the metric's value range to between 0 and 1.
3) **Obstacle Avoidance in Task Space:** This measure indicates whether the user successfully programmed the robot to avoid obstacles placed in the task space for each condition.
4) **Obstacle Avoidance in Joint Space:** Similarly, this measure reflects the success of programming the robot to avoid obstacles in the joint space for each condition.

B. Subjective Measures

1) **NASA Task Load Index (NASA-TLX):** We used the NASA-TLX [27] to measure the subjective task load on participants in each condition. The NASA-TLX is a questionnaire composed of six questions asking participants to rate their experienced task load on six different aspects on a 21-point scale.
2) **System Usability Scale (SUS):** To evaluate the usability of the system in each condition, we used the SUS [28]. The SUS includes ten questions asking participants to rate the system's usability in different aspects on a 10-point scale. An overall score is then calculated. SUS is a well established scale with benchmarks from studies in different types of user interfaces and applications, allowing us to obtain a general indication of system usability relative to other existing interfaces, not limited to AR or robot [29].
3) **After Scenario Question (ASQ):** this three-item questionnaire measures overall ease of task completion, satisfaction with completion time, and satisfaction with support information. It is a commonly used assessment tool for task-based evaluations.
4) **Subjective Mental Effort Question (SMEQ):** To assess the perceived overall experience of programming the robot using our proposed system, we utilized the SMEQ [30]. This single-item questionnaire features a scale from 0 to 150, with nine verbal labels ranging from 'not at all hard to do' (just above 0) to 'tremendously hard to do' (just above 110).

Additionally, participants were asked custom-written questions to rate the ease of each subtask on a 5-point scale. The subtasks we considered are as follows:

1) **Task 1:** defining robot's trajectory
2) **Task 2:** getting the robot to insert a peg in a hole
3) **Task 3:** getting the robot to avoid an obstacle

They were also asked to identify which method they found the most intuitive, preferred and efficient, and to provide feedback on any issues they encountered, along with recommendations for future improvements.

VI. RESULTS

A. Objective Measures

The task completion time was calculated for each participant using both interaction methods. This time was measured from the start of the participant's interaction with the robot until

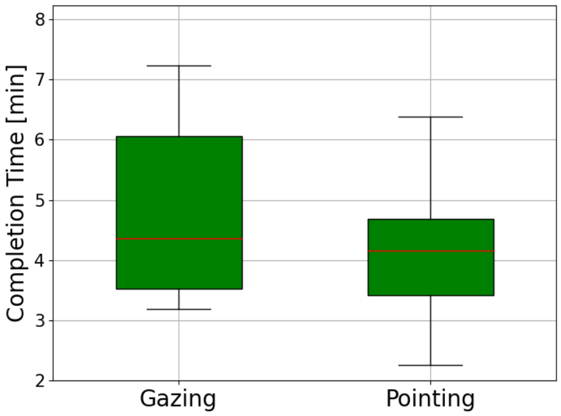

Fig. 6. Task completion time for both interaction methods.

task completion, which involved inserting a peg into two holes and editing the trajectory and joint states to avoid collisions with two obstacles. To ensure accurate time measurement, we excluded periods when the participant was conversing with the experimenter and moments when the experimenter inserted obstacles into the environment. Figure 6 shows a boxplot of the completion times for both modalities. It is shown that participants took slightly longer, 4.83 ± 1.45 min[2] , to complete the task using gazing compared to pointing, which took 4.13 ± 1.07 min. Additionally, there was greater variability in the participants' completion times with the gazing method. A paired t-test indicates that the difference in completion times between the two methods is not statistically significant.

Regarding the task success rate, the gaze method achieved higher success, 0.94 ± 0.17, compared to the pointing method, 0.81 ± 0.31. A paired t-test shows that this difference is also not statistically significant. After adding obstacles to the task and joint space to emulate a real dynamic environment, the obstacle avoidance score for each participant was recorded using both methods. For the obstacle in the task space, only one out of 16 participants failed to edit the trajectory to avoid collision with that obstacle using both methods. Conversely, 6 out of 16 participants did not successfully edit the trajectory in null space to avoid collision with the obstacle in joint space.

B. Subjective Measures

Table I shows the usability scores from NASA-TLX, SUS, ASQ, SMEQ and task ease for both methods. Overall, the pointing method is perceived as more usable than the gazing method. This is shown in the significantly higher SUS score of the pointing than the gaze method. Furthermore, According to the global benchmark for SUS created by Sauro and Lewis, the mean given score is 68 ± 12.5 [29]. Comparing this global mean score from the benchmark with those obtained for our two methods tested, we found that the SUS score for the pointing method is significantly higher than the global mean ($p = 0.01$), while the SUS score for the pointing method is below the global mean for usable systems.

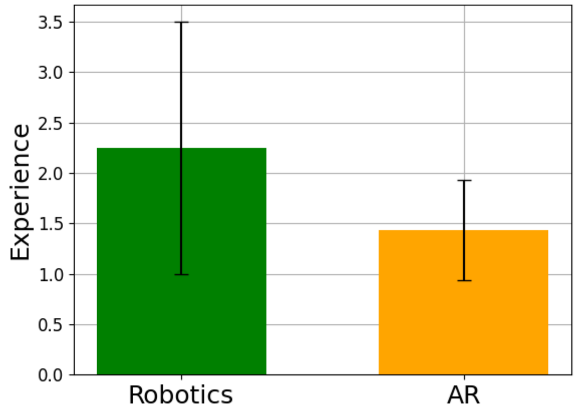

Fig. 7. Previous users' experience in robotics and AR

TABLE I
USABILITY RESULTS FOR NASA-TLX, SUS, ASQ, SMEQ AND MEASURE OF THE EASE OF EACH TASK. METRICS INDICATED WITH (*) PRODUCE A HIGHER SCORE FOR A LOWER USABILITY PERCEPTION.

Metric	Category	Gazing	Pointing	p-value
NASA -TLX (/20)	Mental Demand*	9.69 ± 4.81	7.00 ± 3.97	$p = 0.06$
	Physical Demand*	5.50 ± 2.52	4.81 ± 3.13	$p = 0.4$
	Temporal Demand*	5.44 ± 3.50	4.50 ± 2.83	$p = 0.3$
	Performance	14.94 ± 3.21	15.06 ± 4.23	$p = 0.9$
	Effort*	9.44 ± 4.65	7.31 ± 4.48	$p = 0.06$
	Frustration*	8.94 ± 4.58	4.25 ± 3.33	$p < 0.001$
SUS (/100)	-	60 ± 16	76 ± 11	$p < 0.0005$
ASQ (/7)	Ease of Use	4.56 ± 1.82	5.75 ± 0.93	$p < 0.05$
	Time Satisfaction	4.63 ± 2.00	5.88 ± 1.09	$p = 0.05$
	Provided Support	6.13 ± 0.89	6.31 ± 0.87	$p = 0.19$
SMEQ* (/150)	-	32 ± 22	15 ± 9	$p < 0.01$
Task Ease (/5)	Task 1	3.88 ± 0.89	4.25 ± 0.68	$p = 0.11$
	Task 2	3.75 ± 1.00	3.81 ± 1.11	$p = 0.77$
	Task 3	2.75 ± 1.24	3.31 ± 1.01	$p = 0.057$

Although the satisfaction with the provided support/training is similar between the two methods, the users found that the pointing method is significantly easier to use to complete the task than the gazing method. In addition, users are more satisfied with the amount of time they took to complete the task with pointing than with gazing, which is reflected in both objective and subjective metrics. Furthermore, users after only one trial of both methods, rated their experience with robot programming in SMEQ as *'not very hard to do'* with the pointing method while they rated their experience as *'fairly hard to do'* with the gazing method. Moreover, users showed significantly higher frustration using the gazing method than the pointing. Also, both mental demand and effort needed to complete the tasks using the gazing method are marginally significantly higher than the pointing method.

Figure 7 presents the Likert scale assessments of participants' prior experience in robotics and AR. The data skews towards less or no experience, indicating a novice participant pool. This was an intentional choice to assess the intuitiveness of our proposed system for our target user group. Following the user study, participants were asked to

---

[2] $mean \pm std$

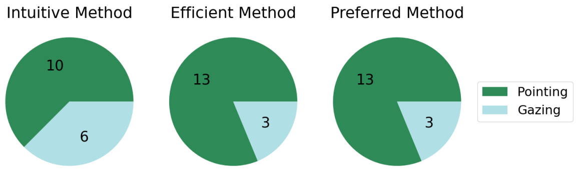

Fig. 8. General feedback on methods preference

identify the method they found most intuitive, efficient, and preferable. Figure 8 summarizes the responses for the users' most intuitive, efficient, and preferred method. It indicates a preference for the pointing method, which most users consider more intuitive and efficient. Only three out of the 16 users indicated their preference for gazing as the most intuitive and efficient method.

Users commented on their choice of the preferred modality, expressing likes and dislikes for each method. Pointing emerged as the preferred method for its intuitiveness and efficiency in robot programming through augmented reality. Users appreciated the clear physical cues and visible interaction lines, which simplified navigation and position interpretation. One user highlighted, *"Seeing the line from my palm told me exactly what I was interacting with"*. Moreover, the pointing method was praised for its quick feedback and precision, especially in tasks like selecting and positioning objects, as another participant noted, *"Hand didn't have as much feedback delay as Gaze did"*.

Conversely, gazing, while considered natural and potentially intuitive, faced issues with consistency, accuracy, and fatigue. *"Gaze actually felt really good when it worked"* a user remarked, pointing out the potential of the method despite its irregular performance. However, this method often proved less efficient, particularly in complex tasks requiring fine adjustments on the robot's joints. Users reported challenges with gaze drift and a lack of depth perception, as evident in the statement, *"Using gaze was not as fast as using the ray from my hand"*. Another user stated, *"The gaze system was a bit strenuous on my eyes"*. Overall, while the pointing method was favored for its direct feedback and precision, gazing was seen as having potential but currently falls short due to its limitations in consistency and precision.

Furthermore, users provided valuable suggestions for improvements in the AR system. Many emphasized the importance of enhancing the system's sensitivity to small eye and hand movements, suggesting that this could alleviate some of the challenges experienced with the current system. A recurrent theme was the need for clearer and less cluttered displays when editing joint movements or avoiding obstacles.

VII. DISCUSSION

The paper proposes an intuitive, AR-based interface aimed at enabling everyday users to program robots easily and efficiently. In our comparative analysis between two interaction modalities, gazing and pointing, we utilized both objective and subjective metrics to identify the most intuitive, efficient, and preferred user modality for an industrial task, peg-in-a-hole. Our results indicated that pointing generally outperformed gazing in terms of task performance and was perceived by users as more intuitive, efficient, and usable.

In terms of task completion time, users took longer with the gazing method (average 4.83 ± 1.45 min) compared to pointing (average 4.13 ± 1.07 min). Furthermore, there was greater variability in completion times with gazing, as shown in Figure 6, suggesting a mixed user experience. Some users found gazing to be a natural way of interaction, while others struggled with inaccuracy and experienced eye strain and fatigue. This aligns with previous findings indicating a potential risk of eye strain or dry-eye syndrome associated with decreased blink rate in gaze-controlled AR interfaces [31], [32].

For obstacle avoidance, both methods achieved similar performance levels. However, a lower number of users were able to successfully edit the trajectory in null space to avoid collisions with obstacles in the robot's joint space. Some users highlighted that gazing was not accurate for selecting the correct robot's hologram, as seen in Figure 5-5, to edit its joint values. Others noted the issue that *'the display gets quite crowded and it can be hard to see which position you're trying to edit'* with both methods, suggesting that refinements in null space control could improve performance in dynamic environments.

Although gazing achieved a slightly higher task success rate, it scored lower in subjective metrics. Users reported difficulty in aligning their gaze with their intended targets, leading to eye strain and fatigue. This is reflected in the higher values of mental demand, effort, and frustration associated with the gazing method, as well as in its lower usability score (60 ± 16) compared to pointing (76 ± 11). Furthermore, users' average response to the SMEQ indicated that robot programming with pointing was *'not very hard to do'* (average score 15 ± 9), while with gazing it was considered *'fairly hard to do'* (average score 32 ± 22), even reaching *'pretty hard to do'* for two participants who wore eyeglasses and struggled with gaze detection inaccuracy. These results underscore the importance of designing AR interfaces that are not only efficient but also ergonomically sound, especially for novice users who may be more susceptible to such challenges.

While this study provides valuable insights, it has limitations that should be addressed in future research. Firstly, evaluating users' performance over multiple trials would be interesting to see if improvements occur over time, especially in terms of efficiency and comfort with each modality. Additionally, further studies involving diverse tasks and user groups could provide a more comprehensive understanding of the ergonomic and cognitive aspects of AR interfaces in robot programming.

VIII. CONCLUSIONS

In conclusion, this research paper has significantly contributed to the field of human-robot interaction by delving into the efficacy of intuitive augmented reality (AR) interfaces for controlling robots among non-expert users. Our study specifically focused on comparing two distinct interaction modalities: gazing and pointing, within the context of AR-assisted robot control. The results derived from the comparative analysis offer compelling evidence in favor of pointing for enhancing the efficiency and user-friendliness of robot programming when tasked with complex operations (i.e., combination of peg-in-hole, trajectory generation, null space control, and obstacle avoidance). These findings provide a practical solution for the development of AR interfaces tailored for human-robot collaboration, especially in industrial settings where non-experts need to interact with robots.

IX. ACKNOWLEDGMENTS

This research was supported in part by a grant of the Korea Health Technology R&D Project through the Korea Health Industry Development Institute (KHIDI), funded by the Ministry of Health & Welfare, Republic of Korea (grant number:HI19C1234), and in part by the Natural Sciences and Engineering Research Council of Canada (infrastructure and operating funds).

REFERENCES

[1] W. Grobbelaar, A. Verma, and V. K. Shukla, "Analyzing human robotic interaction in the food industry," in *Journal of Physics: Conference Series*, vol. 1714, no. 1. IOP Publishing, 2021, p. 012032.

[2] V. Villani, F. Pini, F. Leali, and C. Secchi, "Survey on human–robot collaboration in industrial settings: Safety, intuitive interfaces and applications," *Mechatronics*, vol. 55, pp. 248–266, 2018.

[3] S. Yang, S. Wang, and S. Huang, "Ros-based remote control of industrial robot joystick," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 237, no. 1, pp. 160–169, 2023.

[4] Z. Otarbay, I. Assylgali, A. Yskak, and M. Folgheraiter, "Development of a teach pendant for humanoid robotics with cartesian and joint-space control modalities," in *2019 28th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2019, pp. 1–6.

[5] J. Carmigniani and B. Furht, "Augmented reality: an overview," *Handbook of augmented reality*, pp. 3–46, 2011.

[6] R. K. Khamaisi, E. Prati, M. Peruzzini, R. Raffaeli, and M. Pellicciari, "Ux in ar-supported industrial human–robot collaborative tasks: A systematic review," *Applied Sciences*, vol. 11, no. 21, p. 10448, 2021.

[7] G. d. M. Costa, M. R. Petry, and A. P. Moreira, "Augmented reality for human–robot collaboration and cooperation in industrial applications: A systematic literature review," *Sensors*, vol. 22, no. 7, p. 2725, 2022.

[8] D. Weber, W. Fuhl, E. Kasneci, and A. Zell, "Multiperspective teaching of unknown objects via shared-gaze-based multimodal human-robot interaction," in *Proceedings of the 2023 ACM/IEEE International Conference on Human-Robot Interaction*, 2023, pp. 544–553.

[9] J. Fu, A. Rota, S. Li, J. Zhao, Q. Liu, E. Iovene, G. Ferrigno, and E. De Momi, "Recent advancements in augmented reality for robotic applications: A survey," in *Actuators*, vol. 12, no. 8. MDPI, 2023, p. 323.

[10] A. K. Mutasim, A. U. Batmaz, and W. Stuerzlinger, "Pinch, click, or dwell: Comparing different selection techniques for eye-gaze-based pointing in virtual reality," in *Acm symposium on eye tracking research and applications*, 2021, pp. 1–7.

[11] W. P. Chan, G. Hanks, M. Sakr, H. Zhang, T. Zuo, H. M. Van der Loos, and E. Croft, "Design and evaluation of an augmented reality head-mounted display interface for human robot teams collaborating in physically shared manufacturing tasks," *ACM Transactions on Human-Robot Interaction (THRI)*, vol. 11, no. 3, pp. 1–19, 2022.

[12] H. Bai, P. Sasikumar, J. Yang, and M. Billinghurst, "A user study on mixed reality remote collaboration with eye gaze and hand gesture sharing," in *Proceedings of the 2020 CHI conference on human factors in computing systems*, 2020, pp. 1–13.

[13] M. N. Lystbæk, P. Rosenberg, K. Pfeuffer, J. E. Grønbæk, and H. Gellersen, "Gaze-hand alignment: Combining eye gaze and mid-air pointing for interacting with menus in augmented reality," *Proceedings of the ACM on Human-Computer Interaction*, vol. 6, no. ETRA, pp. 1–18, 2022.

[14] S. Rossi, E. Leone, M. Fiore, A. Finzi, and F. Cutugno, "An extensible architecture for robust multimodal human-robot communication," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 2208–2213.

[15] S. M. Nizam, R. Z. Abidin, N. C. Hashim, M. C. Lam, H. Arshad, and N. Majid, "A review of multimodal interaction technique in augmented reality environment," *Int. J. Adv. Sci. Eng. Inf. Technol*, vol. 8, no. 4-2, p. 1460, 2018.

[16] C. P. Quintero, S. Li, M. K. Pan, W. P. Chan, H. M. Van der Loos, and E. Croft, "Robot programming through augmented trajectories in augmented reality," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 1838–1844.

[17] X. Yan, C. Chen, and X. Li, "Adaptive vision-based control of redundant robots with null-space interaction for human-robot collaboration," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 2803–2809.

[18] J. Garzón, "An overview of twenty-five years of augmented reality in education," *Multimodal Technologies and Interaction*, vol. 5, no. 7, p. 37, 2021.

[19] M. Eckert, J. S. Volmerg, C. M. Friedrich *et al.*, "Augmented reality in medicine: systematic and bibliographic review," *JMIR mHealth and uHealth*, vol. 7, no. 4, p. e10967, 2019.

[20] L. F. de Souza Cardoso, F. C. M. Q. Mariano, and E. R. Zorzal, "A survey of industrial augmented reality," *Computers & Industrial Engineering*, vol. 139, p. 106159, 2020.

[21] H. Sadeghian, L. Villani, M. Keshmiri, and B. Siciliano, "Task-space control of robot manipulators with null-space compliance," *IEEE Transactions on Robotics*, vol. 30, no. 2, pp. 493–506, 2013.

[22] Microsoft, "Microsoft hololens 2," https://www.microsoft.com/en-us/hololens/hardware, 2022, accessed: 2023-08-10.

[23] ——, "Microsoft qr code tracking overview," https://https://wwww.learn.microsoft.com/en-us/windows/mixed-reality/develop/advanced-concepts/qr-code-tracking-overview, 2022, accessed: 2023-09-16.

[24] R. Su, K. Xu, L. Zhao, and P. Yu, "A heterogeneous master-slave teleoperation method for 7-dof manipulator," in *2021 China Automation Congress (CAC)*. IEEE, 2021, pp. 1740–1744.

[25] T. Klamer and S. B. Allouch, "Acceptance and use of a social robot by elderly users in a domestic environment," in *International Conference on Pervasive Computing Technologies for Healthcare*, 2010, pp. 1–8.

[26] M. Moradi, M. Moradi, and F. Bayat, "On robot acceptance and adoption a case study," in *Conference of AI Robotics and 10th RoboCup Iran Open International Symposium*, 2018, pp. 21–25.

[27] S. G. Hart, "Nasa-task load index (nasa-tlx); 20 years later," *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 50, no. 9, pp. 904–908, 2006. [Online]. Available: https://doi.org/10.1177/154193120605000909

[28] J. Brooke, "SUS: A quick and dirty usability scale," *Usability Eval. Ind.*, vol. 189, Nov. 1995.

[29] J. Sauro and J. R. Lewis, *Quantifying the User Experience*. Morgan Kaufmann, 2012. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/C20100651923

[30] J. Sauro and J. S. Dumas, "Comparison of three one-question, post-task usability questionnaires," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. Boston MA USA: ACM, Apr. 2009, pp. 1599–1608. [Online]. Available: https://dl.acm.org/doi/10.1145/1518701.1518946

[31] R. W. Marklin Jr, A. M. Toll, E. H. Bauman, J. J. Simmins, J. F. LaDisa Jr, and R. Cooper, "Do head-mounted augmented reality devices affect muscle activity and eye strain of utility workers who do procedural work? studies of operators and manhole workers," *Human factors*, vol. 64, no. 2, pp. 305–323, 2022.

[32] Y. Wang, G. Zhai, S. Chen, X. Min, Z. Gao, and X. Song, "Assessment of eye fatigue caused by head-mounted displays using eye-tracking," *Biomedical engineering online*, vol. 18, pp. 1–19, 2019.