Creating a Shared Reality with Robots: How Augmented Reality can Facilitate Perception-Related Tasks, Guide Manipulation Behaviors, and Identify Trajectory Failures

A dissertation

submitted by

Andre Vivero Cleaver

In partial fulfillment of the requirements
for the degree of

Doctor of Philosophy

in

Mechanical Engineering:Human-Robot Interaction

TUFTS UNIVERSITY

May 2024

ADVISOR: Dr. Jivko Sinapov

Copyright

by

Andre Vivero Cleaver

2024

Creating a Shared Reality with Robots: How Augmented Reality can Facilitate Perception-Related Tasks, Guide Manipulation Behaviors, and Identify Trajectory Failures

Andre Vivero Cleaver

ADVIS0R: Dr. Jivko Sinapov

Augmented reality (AR) has emerged as a transformative tool in reshaping the field of human-robot interaction (HRI), especially in tasks requiring seamless manipulation and precise pick-and-place actions. This dissertation explores the multifaceted utilization of AR devices in enhancing user comprehension and performance across a range of robotic activities. Our initial study investigates the deployment of AR devices to gauge user engagement and performance in robot training and education. By visualizing robots' sensory and cognitive representations within their surroundings, AR facilitates intuitive validation and troubleshooting of internal robot processes. Results indicate noticeable enhancements in user performance, particularly in tasks involving object detection and safety, highlighting the significance of AR in first robot encounters. Our next study explores the collaborative dimension of HRI within fluid manipulation tasks, where human intuition often surpasses robotic capabilities. By engaging users in correcting and providing feedback on planned pouring trajectories through AR visualization, the study demonstrates substantial reductions in spills. Both AR headsets and conventional 2D displays demonstrate the potential of a shared reality space in complex tasks. Lastly, our research examines the effectiveness of visualized information in AR for identifying failures in robotic pick-and-place tasks. Findings revealed the trade-offs

between static and animated representations, offering insights into the balance between levels of trajectory information conveyed and visual density for enhancing user experience and task performance. In summary, this dissertation contributes to advancing our understanding of AR's role in HRI, providing valuable insights into its application across diverse robotic tasks. By understanding the benefits and challenges of AR integration with robotic systems, this research lays the groundwork for the development of more efficient and intuitive AR systems across various robotic domains.

Acknowledgments

First and foremost, I would like to thank my advisor, Jivko. Thank you for your mentorship, your patience, your teachings, and thank you for showing me AR. I still remember the day when I first entered your office as a lost student, and you showed me what would be my first experience with AR. Ever since that day, I've learned how to make really cool AR robot demos!

I'd like to thank my dissertation committee members Prof. Elaine Short and Prof. Jason Rife for their time, their mentorship, teachings and advising throughout the years. I would also like to thank my dissertation committee member Dr. Aldis Sipolins, for hiring me at Draper in the summer of 2021. It was from that co-op position that I got more exposure to new styles of AR and learned new tricks in Unity.

I'd like to thank Reuben Aronson for your valuable feedback towards my HRI studies. I'd like to thank Brennan for your technical contributions toward my HRI studies. I'd like to thank former students: Darren Tang, Victoria Chen, and Naomi Arnold for their help with my HRI studies.

I would like to thank the GEM Foundation for funding my first year of my Ph.D. and for selecting me for my first summer internship at MIT Lincoln Labs. I wouldn't have been able to work with Jivko otherwise.

I would like to also thank other various organizations that funded my projects throughout the years: Verizon Foundation, Amazon, Tufts' Center for Applied Brain and Cognitive Sciences (CABCS), PTC, and the National Science Foundation (NSF). I would also like to thank the Black in Robotics organization for their support and travel scholarship to fund my HRI conference.

I'd like to thank my colleagues in the MuLIP Lab, AABL Lab, and the HRI

Lab, both past and current students. It's the conversations with you all that help spark new ideas, clarify ideas, and bring reassurance.

I'd also like to thank my Master's Advisor, Douglas Matson from the Mechanical Engineering department who helped with my transition to the Ph.D. program even after telling him I don't want to do a Ph.D. in material science.

I would like to thank my parents for their support and many care packages, and I would like to thank my aunties for helping me with my transition to Boston from Texas. I'd like to thank Rob, my therapist for the last 4 years, for helping me see the big picture and focus on what I want for my life. And just as Snoop Dogg puts it best, "I wanna thank me. I wanna thank me for believing in myself, I wanna thank me for working so hard, I wanna thank me for never quitting. . . even though I've checked out a long time ago. I wanna thank me for staying me." And to all my friends I've made along the way, thanks for the laughs, I needed them.

Andre Vivero Cleaver

TUFTS UNIVERSITY

May 2024

Contents

Abstract ii

Acknowledgments iv

List of Tables xii

List of Figures xiii

Chapter 1 Introduction 1

1.1 Augmented Reality in Human-Robot Interaction 1

1.1.1 Shared-Reality Space 3

1.1.2 Design Space 5

1.2 Research Questions 7

1.2.1 In which cases do people want to use Augmented Reality when interacting with a robot? 8

1.2.2 How can AR be leveraged to help with predictions in complex robotic interaction scenarios such as Fluid Manipulation? . . 8

1.2.3 How can robotic information relating to trajectory be visualized effectively in AR to showcase robotic plans to ensure that individuals can comprehend them efficiently? 9

1.2.4 Contributions 9

1.2.5 Overview 11

Chapter 2 Background & Related Work 12

2.1 Extended Reality (XR) . . . 13
2.2 Theory of Mind and Robots . . . 15
2.2.1 Human Internal Physics Engine . . . 16
2.3 Human-Robot Interaction: Nonverbal Communication . . . 17
2.3.1 AR for HRI . . . 19

Chapter 3 Systems Description **20**

3.1 Hardware . . . 20
3.1.1 Turtlebot2 . . . 20
3.1.2 Kinova Gen3 Lite . . . 22
3.1.3 Microsoft Hololens2 . . . 22
3.1.4 Samsung Galaxy Tab s8 Tablet . . . 25
3.1.5 Mechanical button for Detector-Response Test . . . 25
3.2 Software . . . 26
3.2.1 Robot Operating System (ROS) . . . 26
3.2.2 Unity . . . 26
3.2.3 Vuforia . . . 27
3.2.4 Database Server for Data Collection . . . 29
3.3 Software: SENSAR[§§] . . . 30
3.3.1 Introduction . . . 30
3.3.2 Related Work . . . 31
3.3.3 SYSTEM ARCHITECTURE AND DESIGN . . . 33
3.4 Software: RAIN[§§] . . . 36
3.4.1 Proposed Study . . . 42

[§§]**This chapter is based on the following paper:** Andre Cleaver, Faizan Muhammad, Amel Hassan, Elaine Short, and Jivko Sinapov. SENSAR: A shared reality with intelligent robots for collaborative human-robot interaction. In AAAI Fall Symposium Series: Trust Explainability in Artificial Intelligence for Human-Robot Interaction (AI-HRI), 2020

[§§]**This chapter is based on the following paper:** Andre Cleaver, Elaine Short, and Jivko Sinapov. RAIN – A vision calibration tool using augmented reality. In ACM/IEEE Conference on Human-Robot Interaction (HRI), Workshop on Virtual, Augmented and Mixed Reality for Human-Robot Interaction (VAM-HRI), Cambridge, UK, 2020

3.4.2 Outcome Measures . . . 43
3.4.3 Procedure . . . 44
3.4.4 APPLICATIONS . . . 45
3.4.5 CONCLUSIONS . . . 46
3.5 Software: HAVEN[§§] . . . 47
3.5.1 Introduction . . . 48
3.5.2 RELATED WORK . . . 50
3.5.3 SYSTEM ARCHITECTURE AND DESIGN . . . 50
3.5.4 APPLICATIONS . . . 54
3.5.5 DISCUSSION . . . 56
3.5.6 CONCLUSION . . . 56
3.5.7 HRI Study: Dynamic Path Visualizations[§§] . . . 57
3.5.8 METHODOLOGY . . . 60
3.5.9 Discussion and Future Work . . . 68
3.6 Summary . . . 70

Chapter 4 First Encounters with a Robot: The Value of Augmented Reality for Non-Experts Performing Multiple Tasks with a Mobile Robot[§§] 72

4.1 Introduction . . . 72
4.2 Related Work . . . 74

[§§]**This chapter is based on the following paper:** HAVEN: A unity-based virtual robot environment to showcase HRI-based augmented reality. In AAAI Fall Symposium Series: Trust Explainability in Artificial Intelligence for Human-Robot Interaction (AI-HRI), 2020

[§§]**This chapter is based on the following paper:** Andre Cleaver, Darren Tang, Victoria Chen, and Jivko Sinapov. Dynamic path visualization for human-robot collaboration. In ACM/IEEE International Conference on Human-Robot Interaction (HRI) Late-Breaking Report, 2021

[§§]**This chapter is based on the following paper:** Andre Cleaver, Victoria Chen, and Jivko Sinapov. First Encounters with a Robot: The Value of Augmented Reality when Learning about Mobile Robots. Accepted to the ACM/IEEE International Conference on Human-Robot Interaction, Workshop on Virtual, Augmented and Mixed Reality for Human-Robot Interaction (VAM-HRI 2023

4.3 Methodology . . . 75
4.3.1 Hardware and Software . . . 76
4.3.2 Users and Procedures . . . 76
4.3.3 Robotic Activities . . . 78
4.3.4 Measures & Analysis . . . 81
4.4 Results . . . 82
4.5 Discussion . . . 87
4.6 CONCLUSIONS and FUTURE WORK . . . 89

Chapter 5 Helping People Predict the Outcome of Robotic Pouring Behaviors with Augmented Reality[§§] 92

5.1 Introduction . . . 92
5.2 Related Work . . . 95
5.3 Methodology . . . 97
5.3.1 Procedure . . . 98
5.3.2 Pouring Parameters . . . 101
5.3.3 Conditions . . . 101
5.3.4 Participants . . . 103
5.3.5 Measures & Analysis . . . 105
5.3.6 Hardware & Software . . . 107
5.4 Results . . . 108
5.5 Discussion . . . 110
5.5.1 Limitations & Future Work . . . 114
5.6 Conclusions . . . 115

Chapter 6 Selecting Acceptable Robotic Pick-and-Place Plans with Varied Information in Augmented Reality 116

[§§]**This chapter is based on the following paper:** Andre Cleaver, Reuben M. Aronson, and Jivko Sinapov. Helping People Predict the Outcome of Robotic Pouring Behaviors with Augmented Reality. Accepted to the ACM/IEEE International Conference on Human-Robot Interaction, Workshop on Virtual, Augmented and Mixed Reality for Human-Robot Interaction (VAM-HRI 2023

6.1 Introduction . . . 116
6.1.1 Research Questions . . . 118
6.1.2 Hypotheses . . . 118
6.2 Related Work . . . 119
6.3 Methodology . . . 121
6.3.1 Visual Conditions . . . 121
6.3.2 Failure Conditions . . . 122
6.3.3 Ground Truth Pick-And-Place Plans . . . 123
6.3.4 Participants . . . 124
6.3.5 Task Description . . . 125
6.3.6 Open-Condition Task . . . 126
6.3.7 Measures & Analysis . . . 126
6.3.8 Hardware & Software . . . 128
6.4 Results . . . 128
6.5 Discussions . . . 137
6.6 Limitations & Future Work . . . 142
6.7 Conclusions . . . 143

Chapter 7 Future Work **145**

7.0.1 What information of a robotic reinforcement learning (RL) problem do people benefit the most when visualized in AR? . 146
7.0.2 Enable modifications of a robotic scene to better improve the robot's learning process based on visualizations determined in the first study? . . . 146
7.0.3 Development of a dynamic visualization algorithm that is adaptive in visualizing concise information for training a robot in context of a reinforcement learning problem . . . 146
7.1 TRAinAR[§§] . . . 147

[§§]**This chapter is based on the following papers:** Andre Cleaver, and Jivko Sinapov. Helping Humans Become Better Teachers for Robots with Augmented Reality. Accepted to the ACM/IEEE International Conference on Human-Robot Interaction (Workshop VAM-HRI 2023)

7.2 Related Work 148
7.3 Methodology 149
7.3.1 Task Descriptions 151
7.3.2 Hypotheses 152
7.3.3 Hardware & Software 153
7.3.4 The Virtual Training Environment 153
7.3.5 Measures & Analysis 154
7.4 Discussion 155

Chapter 8 Conclusions **156**
8.1 Lessons & Take Aways 157
8.2 Summary 159

Bibliography **160**

and Andre Cleaver, and Jivko Sinapov. Demonstrating TRAinAR: An Augmented Reality Tool that Helps Humans Teach Robots. Accepted to the ACM/IEEE International Conference on Human-Robot Interaction (HRI 2023)

List of Tables

5.1 Configuration table for each source container. The research coordinator references this chart to ensure the correct parameters are set prior to each participant. 99

5.2 Visual and Container order for all participants. * indicates a container that required correction to prevent spill. 104

List of Figures

3.1 Image of the Turtlebot2 robot used in this dissertation. The QR-printed cube attached to the top of the robot serves a target image for tracking purposed discussed further in 3.2.3 21
3.2 Image of the RPLIDAR sensor and Astra camera mounted to the Turtlebot2 that were used in this dissertation. 22
3.3 Image of the Kinova Gen3 Lite robot used in this dissertation. Towards the base of the arm is a cylindrical target image used or tracking the robots pose. See section 3.2.3) for further details. 23
3.4 Image of the Microsoft Hololens2 used in this dissertation. 24
3.5 Image of the mechanical button used in Chapter 6 for the Detector-Response Test. 26
3.6 Image of the Tablet System used in this dissertation. (Top Left): Mounting arm, (Top Right): Samsung Galaxy Tab s8, (Bottom): Tablet mounted to table 27
3.7 Image of a screenshot on a Samsung Galaxy Tab s8 running a Unity-based AR application to visualize robot data. The left panel includes a list of robot data visualization options that a user can toggle, while the right shows the robot and the corresponding robot data visualized. 28

3.8 Image of a robot with a Multi-Target image tracker fixed on top of a Turtlebot2 robot for tracking. Once tracked, AR visuals programmed in Unity can be rendered by the AR device. This figure shows various types of visual data you can display with in AR. (A) Localization, (B) Laser scan, (C) People Detection, (D) Cost Map. 29
3.9 Interactive virtual space shared by a human and a mobile robot. The Hololens renders the robot's Laser scan data (red markers on human), Occupancy-Grid data (red/green dots in grid formation), and planned path trajectory (red/yellow markers). 30
3.10 (Right) Physical world known by a human; robots are unaware of most objects within environment, (Left) Virtual World accessed only by the robot what includes the robot's algorithms and sensor readings, (Center) a shared reality world accessed by both humans and robots. A robot renders its path plan using markers which is seen by human. 32
3.11 System Architecture - Our system communicates various data types between the Robot and AR device through a web-socket connection. Sensor data collected through nodes go through a filtering layer for down-sampling and to remove non-numerical data points (i.e. NaN). 33
3.12 This figure shows the first design of the robot and the types of visual data you display with AR. (A) Localization, (B) Laser scan, (C) People Detection, (D) Cost Map. 35
3.13 Visualization projected by the robot to indicate intended direction of motion around an obstacle. Top image is path trajectory using yellow markers. Bottom image is turn signals which operates similar to turn signals of a motor vehicle. 37
3.14 Uniform distribution of rendered particles that correspond to the filtered output of the raw data. (Left) Low density and (Right) High density. 42

3.15 Red particles rendered over a detected real-world object. The particles correspond to points in the raw data that fit a cylinder model of a given radius. A user adjusts the radius parameter to detect the object. 44
3.16 Planned path trajectory (yellow markers) with operating laser scan (Red markers) resulting in obstacle avoidance (top row). Robot operating with unexpected malfunctioning laser scan leading to wall collision (bottom row). laser scan did not detect the wall as the robot planned a right turn in image E. The lack of red markers in image E but present in the following image F provided insight that the laser scan is the source of the robot's behavior. 46
3.17 EV3 robot is projecting its ultrasonic sensor data by visualizing the blue cone that extends out towards the object detected by the robot's sensor. Other sensor can be visualized as seen on the right-hand panel. 47
3.18 An AR visualization of the robot (Turtlebot2)'s path. White cubes represent the arc motion while the white spheres show straight motion. 48
3.19 Examples of AR visualizations projected by the Turtlebot2 robot. Thought bubble is rendered on top. Battery indicator is in green in the middle, and the yellow arrow are used for motion intent. 51
3.20 First person view of the hallway passing scenario. The TurtleBot2 detects the human (obstacle) blocking its path and therefore re-plans its path trajectory to the goal. The direction it decides to move around the obstacle is given by the rendered yellow arrow to communicate its motion intent. 54
3.21 First person view of object retrieval scenario. The red circle on the floor is the TurtleBot2's charging base. The white spheres show the TurtleBot2's path trajectory. Blue gems are robot collectibles. Red gems are human collectibles. 55

3.22 Screenshot of participant's first-person-view of a robot (Turtlebot2) projecting its path trajectory (Blue arrows) and game objects (red gems) that must be collected. Participant's progress and score are shown in text above. 57
3.23 Path visual conditions from left to right: **None**, green line shows the path of the robot with no arrows, **Continuous**, blue arrows are shown along the entire path, **Fixed**, a specified amount of time is shown constantly, and **Dynamic**, a function that uses a natural logarithmic equation that takes the path's tortuosity as input to calculate the distance ahead of the robot's path to reveal. 59
3.24 Subjective results: Averaged NASA-TLX scores (Left), Averaged ratings for Clarity, Helpfulness, and Safety(Right) 65
3.25 Average Time completion for each condition. First letter indicates condition, second letter indicates path condition, and the number indicates round. 67
3.26 Average number of contacts made on either side of robot for each condition. 68

4.1 Screenshot of *SENSAR* on a tablet. The application detects the robot's pose with the target image cube and renders the selected robotic data types in relation to the cube. 73
4.2 Visualizing different robotic data set types shown in RViz (top) and through an AR device (bottom). From left to right, Laserscan (Sensors), Person Detector (Cognitive), Battery Levels (Diagnostics), Safety-Zone (Safety), and Goal Point (Actions) 74
4.3 The lab space where the experiment took place with robot in the center. Objects in scene were part of the Object Detection task. The Control group used a desktop computer that remained to the side of the room (left side of image). 77

4.4 Figure shows Info Mat reference sheet provided for participants that highlights sections of RViz. 78
4.5 Plot of the average accuracy between the the two groups within the Object Detection Task. Solid bars indicated standard deviation. . . 83
4.6 Plot of the average confidence range between the two groups within the Safety activity. Solid black bars represent the standard deviation 84
4.7 Plot of average time spent on visualization tools among the Test group for each task. Solid black bars represent the standard deviation. . . 85
4.8 Plot of average 7 point Likert-Style on user preference towards the AR device for each robotic activity within the Test group. Solid black bars represent standard deviation. 86
4.9 Plot of average QUAED ratings within the Test group. Asterisks indicate significant differences. 87

5.1 Visualization of the three conditions in our study: (Left) No Visuals/Control(CT), (Middle) RViz/2D Display(RZ) with a model of the environment, (Right) AR from the prospective of the participant. Screen capture from a Hololens2. The robot and source container digital twins are superimposed over the physical robot and container respectively. 97
5.2 Screenshot of the control panel used to adjust the parameters of the robot's pouring behavior 101
5.3 The 10 source containers used in this study. 102
5.4 Average percentage fluid transfers by visual condition. T-bars indicate standard error. 107
5.5 Average score in *Parameter Selection* by condition (Left) and by container type (right) where N and W correspond to Narrow and Wide respectively. 110

6.1 Screenshots from the point of view of a participant wearing a Hololens2 observing a robotic plan shown in the three visual conditions. The robot plan demonstrates picking up the silver cup and placing it on the adjacent tray table (Left) *End Points* condition. (Middle) *Key Frames* condition (Right) *Animated* condition. 119
6.2 Screenshot from the tablet device where the participant answers questions regarding the current plan they are evaluating. 124
6.3 Accuracy Score, Confidence rating, and Time to Evaluate of No Failure (**: 1.00e-3 ¡ p ¡= 1.00e-2) (***: 1.00e-4 ¡ p ¡= 1.00e-3) (****: p ¡= 1.00e-04) 129
6.4 Accuracy Score, Confidence rating, and Time to Evaluate of Directional Path Failure (**: 1.00e-3 ¡ p ¡= 1.00e-2) (***: 1.00e-4 ¡ p ¡= 1.00e-3) (****: p ¡= 1.00e-04) 129
6.5 Accuracy Score, Confidence rating, and Time to Evaluate of Social Failure Plan (**: 1.00e-3 ¡ p ¡= 1.00e-2) (***: 1.00e-4 ¡ p ¡= 1.00e-3) (****: p ¡= 1.00e-04) 130
6.6 Accuracy Score, Confidence rating, and Time to Evaluate of Grasp Failure. (***: 1.00e-4 ¡ p ¡= 1.00e-3) (****: p ¡= 1.00e-4) 130
6.7 Accuracy Score, Confidence rating, and Time to Evaluate of Placement Failure. (***: 1.00e-4 ¡ p ¡= 1.00e-3) (****: p ¡= 1.00e-4) . . . 131
6.8 Results from the NASA TLX survey. Plot shows the average response value [0-10] for each NASA-TLX category. AN = Animated, KF = Key Frames, EP = End Points. *: 1.00e-02 ¡ p ¡= 5.00e-02, **: 1.00e-03 ¡ p ¡= 1.00e-02, ***: 1.00e-04 ¡ p ¡= 1.00e-03 133
6.9 Average Detector-Response Test (DRT) scores across visual conditions. Bars indicate standard error. * indicates a marginal difference between *Key Frames* and *Animated*, ($p = .049$) 134
6.10 Average time in minutes to complete trajectory evaluation task across conditions. 135

6.11 Percent Average of time using each visual condition to complete trajectory evaluation across conditions. (****: p ¡= 1.00e-04) 136

7.1 Kinova Gen3 Lite robot with augmented digital twin conveying motion intent (purple arm) with target goal pose (yellow arrow). 148

7.2 Kinova Gen3 Lite robot with virtual boundary planes (ground and ceiling) that restricts both the robot's movement as well as the exploration space. Users can physically interact with both planes to constrain the environment to their preference. 150

7.3 The user visualizes the training data history over the real-world environment. The agent here is represented as a blue cube which spawns in random locations for each training episode. Users with visualized training data can gain insights into the agent's learning performance. 152

Chapter 1

Introduction

1.1 Augmented Reality in Human-Robot Interaction

Communicating effectively with non-humanoid robots poses a significant challenge, as they lack the natural communication features inherent in human-like designs [23]. While humans can convey information non-verbally through eye gaze, gestures, facial expressions, and whole-body movements [98], these cues present difficulties for even the most advanced non-humanoid robots. Additionally, when encountering a new or unfamiliar robot, individuals, regardless of their level of expertise in robotics, may rely on the design principle of *"Form follows Function"* to infer the robot's purpose from its appearance [101]. However, accurately assessing a robot's purpose based solely on visual observation can be challenging. Moreover, determining the robot's current state and identifying potential issues, such as faulty sensors or mechanical problems, is practically impossible through visual inspection alone. Effective human-robot interaction hinges on the robot's ability to communicate its capabilities and status to humans [60]. By conveying this information, the robot not only updates the human's perception but also builds trust in its abilities [129]. Awareness of the robot's functions and limitations empowers humans to navigate interactions with the robot more effectively, mitigating undesired outcomes.

Augmented reality (AR) technology has emerged as a transformative tool in redefining human-robot interaction (HRI), particularly in tasks requiring precise

manipulation and intuitive collaboration [71]. While AR has shown promise in enhancing user understanding and performance across diverse robotic activities, its full potential in augmenting human-robot interaction remains to be explored [174]. This dissertation seeks to investigate the role of AR in enhancing user performance across a range of robotic tasks, with a focus on object perception and manipulation.

It's essential to acknowledge that AR isn't a universal solution for all robotic-related scenarios. Despite its demonstrated benefits, there remains a notable gap in research regarding specific cases where individuals seek to utilize AR within robotics. The existing studies have primarily focused on its effectiveness in perception enhancement, training, and simulation. However, the breadth of potential applications for AR in robotics suggests the need for comprehensive investigations into its suitability and limitations across diverse contexts. Thus, while AR holds promise in augmenting certain aspects of robotic systems, further research is necessary to explore its applicability in a broader range of use cases and to understand when and where its integration may yield the most significant advantages.

AR also encounters certain constraints, for example, in domains where simulations face constraints or intricacies, such as the manipulation of fluids. A notable instance arises in tasks like directing a robot to pour a drink, where the complexity lies in the robot's inability to comprehensively anticipate every nuanced aspect to prevent spillage. Thus, delving deeper into the potential of AR to aid individuals in guiding robots through such tasks, like pouring, becomes imperative. AR can offer real-time observation capabilities, empowering users to monitor dynamic changes and apply their expertise to fine-tune the robot's actions preemptively. This approach not only enhances the collaboration between humans and machines but also leverages human intuition and adaptability to optimize task execution.

Exploring the presentation of forthcoming robotic trajectory plans within augmented reality (AR) holds significant merit, as it strives to equip users with comprehensive information necessary for informed decision-making while maintaining a balance to prevent overwhelming them with excessive visual data that could potentially disrupt their workflow. Current approaches, including the area of virtual

alterations [174] to a robot's morphology, are vital considerations in this endeavor. Specifically, starting with a full digital twin of a robot means that any removal of parts of the robot is categorized as diminished reality [165]. This approach not only contributes to the solution for presenting information that is necessary but also ensures it does not obstruct the user's view of the environment. By delving into this area, we aim explore potential approaches of diminishing reality in the context of robotic manipulation, ensuring that users receive pertinent trajectory information in a clear and digestible manner. This approach not only enhances user understanding of robotic actions but also fosters smoother human-robot interaction, ultimately leading to more efficient and productive task execution. Through careful investigation and refinement, AR has the potential to serve as a seamless conduit between users and robots, facilitating enhanced collaboration and productivity across various domains.

1.1.1 Shared-Reality Space

In this dissertation, we introduce the concept of a shared reality space, where both humans and robots can access and align their perceptions of the environment. In the physical world, human observations often dominate due to their superior sensory abilities compared to robots, which are limited by sensor readings and object detection algorithms. Conversely, the virtual world is primarily controlled by robots, housing their perceptions, decision-making processes, and knowledge. Augmented Reality (AR) serves as a bridge, granting both humans and robots access to this shared reality space. Here, the robot visualizes its world for the human, who can then provide feedback to ensure mutual understanding. Given the challenge many non-humanoid robots face in conveying information in familiar ways, AR offers a potential solution by presenting graphical images in the context of the environment, facilitating intuitive communication.

With the creation of this shared reality space, several questions arise. Firstly, what information is essential to display within this space? Robotic information intended for people can be categorized into several classes. For simplicity, we can de-

termine which of these classes do people find more beneficial when viewed through augmented visuals. While AR technology excels at rendering robotic information, it's crucial to consider the relevance of information from the user's perspective. Once the pertinent information is identified, the challenge shifts to our second question, what to display within this space? While providing users with comprehensive information is desirable, excessive visualizations may overwhelm them. Conversely, simplifying visuals, such as using points or lines, risks depriving users of necessary information. Thus, striking a balance in visual renderings is essential to furnish users with sufficient information for informed decision-making without overwhelming them mentally. Finally, to evaluate the overall effectiveness of the shared reality space, we selected a scenario wherein a robot performs a task complex for itself but trivial for humans. Here, both parties can access the shared reality space, enabling the robot to visualize its planned actions while receiving feedback or corrections from the user.

As a result, our motivation led us to design three distinct human-robot interaction studies, each aimed at addressing key questions. Our initial study was crafted to gauge participants' preferences in utilizing Augmented Reality (AR) across five unique robotic-related activities, each centered around different types of robotic information. Subsequently, we conducted a study to assess the efficacy of our proposed shared reality space through a task uniquely suited to this environment. We selected a pouring task as it exemplifies a scenario where access to the virtual space is imperative for the user, while the robot's ability to perceive its surroundings is challenged. In such instances, the robot relies on the shared reality space to convey vital information, such as its pouring technique, enabling the user to make informed decisions and potentially correct any undesired outcomes. Our final study aimed to determine the balance of information exchange within the shared reality space, particularly focusing on the visual representation of this information. To explore this, we simulated a scenario involving a robotic waiter delivering a cup of water, highlighting situations where the robot's uncertainty necessitates human assistance. By visualizing its delivery plans within the shared reality space, the robot allows

users to assess the acceptability of each plan and provide feedback accordingly.

Overall, this dissertation seeks to investigate the role of AR in enhancing user performance across a range of robotic tasks, with a specific focus on object perception and manipulation. While acknowledging that AR isn't a universal solution for all robotic-related scenarios, it's crucial to delve deeper into its potential within robotics. AR encounters constraints, especially in domains where simulations face intricacies such as fluid manipulation. Delving deeper into the potential of AR to aid individuals in guiding robots through such tasks, like pouring, becomes imperative. Through careful investigation and refinement, AR has the potential to serve as a seamless conduit between users and robots, facilitating enhanced collaboration and productivity across various domains.

1.1.2 Design Space

Augmenting robotic data within the shared reality space typically initiates with a robot publishing a message containing information like its observations, internal states, or motion planning. Following this step, a subscriber, listening to these messages, undertakes the task of filtering them, aiming to condense data size and transmit pertinent information before delivering it to the AR device for visualization. The AR device intercepts these filtered messages, effectively translating them into visual representations, whether by updating existing virtual object states like robot digital twin models or generating new virtual objects, such as particle systems, to depict robotic sensor readings.

The visualization techniques employed are depend on the types of robotic messages received. In this dissertation, trajectory-like messages were predominantly used. These messages typically comprise an array or list of data points, each encompassing a location, a value, and a time period. The location might denote a spatial coordinate or reference a specific robot component, such as a joint on an arm manipulator. The value could encompass various measurements, such as sensor readings, while the time period indicates the duration from the trajectory plan's initiation. For instance, within a joint-state message containing numerous data points,

one might find information indicating that the robot's shoulder joint is positioned at 90 degrees ten seconds into the trajectory plan.

Visualizing trajectory-like messages in our shared reality space requires a design space to suit the needs of the user. The following summarizes the design space for the robotic augmented reality system used for this dissertation, where each definition can be treated as an adjustable parameter:

1. **Spatial Resolution** - This parameter delineates the spacing distance between each rendered point along a trajectory. It signifies the density of the visual representation, detailing how closely points are plotted in space. The range for this axis spans from the maximum distance, where only terminal positions are rendered, to the minimum distance, where all points within the trajectory message are visually depicted, offering a comprehensive view of the path.

2. **Temporal Resolution** - This parameter delineates the degree to which a path trajectory is unveiled in space, encapsulating the concept of "how far into the future do we display the path?" The spectrum for this axis spans from a minimum distance of 0, signifying the absence of visual representation, to the maximum distance, where the entirety of the trajectory message is disclosed, revealing the complete path. For finer control within this axis, trajectory data can be rendered in defined time intervals. For instance, at 5-second intervals, data points falling within each interval are displayed for a corresponding duration, ensuring a systematic visualization of the trajectory's progression.

3. **Speed Factor** - This parameter regulates the pace at which the trajectory message is played back, determining the speed at which successive data points are rendered. This axis encompasses a spectrum ranging from real-time playback to a scaled factor, enabling adjustments to playback speed, whether to accelerate the visualization or even rewind the trajectory.

4. **Visualization Placement** - This parameter governs the spatial arrangement

of rendered data, allowing for the adjustment of its location in 3D space, with the aim of preventing obstruction of the user's field of view. This axis presents a multifaceted control mechanism, granting users the ability to manipulate data within the three-dimensional realm. For instance, spatial coordinates from a trajectory message can be condensed onto one or two axes, thereby visualizing data along a 2D plane or a single axis. Additionally, users can exercise further control by positioning visualizations either along the ground floor or above a robot for enhanced convenience and clarity.

Turning these parameters can enable more visualization patterns such as mixing spatial and temporal resolution where as you project more further into the future, you use less spatial resolution.

Now for each point rendered in space, the following describes the aesthetic design space in representing the visualized data:

1. **Model** - This variable represents the virtual object used to indicate the date point in space. This axis can range from abstract shapes to high resolution digital twin models.

2. **Size** - This variable describes the scaled size of the models themselves.

3. **Color (RGBA)** - This variable describes the material appearance of the models. The axis ranges within the Red, Green, and Blue color wheel and the Alpha value for transparency. Additionally, textures can also be applied to models.

4. **Brightness** - This variable describes the brightness of the models. This variable's range can depend on the devices specifications.

1.2 Research Questions

The main research question that is investigated in this dissertation is the following:

How can augmented reality technology enhance human-robot interaction and improve user performance across a range of robotic tasks?

More specifically, this question is addressed by breaking it into the 3 subsidiary questions that are listed below:

1.2.1 In which cases do people want to use Augmented Reality when interacting with a robot?

AR devices for robot training and education have introduced people to a new perspective in robot interaction [29, 64].

For instance, users can validate or debug a robot's internal representations and environmental perceptions as well as explore the robot's internal state and capabilities [117, 35]. Although AR devices in previous human-robot interaction studies have assisted users in mostly single robotic tasks, it is unclear in what settings users will want to utilize an AR device across multiple types of interactions to accomplish their tasks. Specifically this paper aimed to understand how users use AR, what value it brings, and in which types of robotic-related tasks it is most useful. The experimental results described in Chapter 4 show that people preferred using an AR device in activities that relate to Object Detection and Safety. The results also show that users with access to an AR device significantly performed better compared to users without in the same categories.

1.2.2 How can AR be leveraged to help with predictions in complex robotic interaction scenarios such as Fluid Manipulation?

Similar to studies in conveying robot motion intent, the research described explores how AR can be used to guide a robot's behavior in a more complex scenario often viewed as trivial for humans but rather complex for robots. People effortlessly manipulate fluids due to their intuitive understanding of fluid dynamics, while robots struggle with complex fluid dynamic calculations, particularly in tasks like pouring. To enhance assistive robots in such tasks, we propose involving users in correcting

and providing feedback by visualizing the planned pouring trajectories before they are executed. Chapter 5 investigated whether people can predict robotic pouring outcomes and make adjustments to minimize spills, using visualization devices like augmented reality. Results highlight the effectiveness of visualization tools such as augmented reality headsets, as well as traditional 2D display, especially with specific pouring parameters.

1.2.3 How can robotic information relating to trajectory be visualized effectively in AR to showcase robotic plans to ensure that individuals can comprehend them efficiently?

The effectiveness of visualized information in augmented reality (AR) was evaluated for users to identify failures in robotic pick-and-place tasks, specifically those involving the delivery of a cup of water. A human-participant study was conducted to assess users' ability to discern the outcomes of robotic plans when observed through AR. Three visual conditions were created that offer varying levels of information regarding the robots plans: End Points, Key Frames, and Animated representations. Findings described in Chapter 6 suggest that while both End Points and Key Frames provided quick trajectory information, they struggled to convey robot speed effectively. Conversely, users detected failures related to poor path directions better with Key Frames and Animated representations. Preference for Animated representations was evident, despite increased time investment. However, Key Frames risked overwhelming participants and accuracy. Balancing static and animated renderings is crucial, offering insights into trajectory information levels for robotic tasks involving fluids and nuanced trade-offs between visual conditions.

1.2.4 Contributions

The main contribution of this work is greater insights into the uses and limitations of Augmented Reality in the context of Human-Robot Interaction. In particular, determining 1) what particular cases do people want robotic data visualized in AR,

2) how do you present robotic trajectory efficiently in AR for people to understand, and 3) how can AR improve user prediction from robotic actions. The studies presented in this work have resulted in new techniques and suggestions for improving future HRI studies. This work also contributes to research towards AR design interfaces for robot interactions.

The following are the primary contributions of this dissertation:

1. An AR-application that visualizes multiple types of robotic data for HRI studies that determined user preference towards augmented reality. (Chapter 4)(Chapter 5)
2. An AR-application that visualizes real-time robot pouring parameter changes to enable user-guided control for fluid manipulation and spill reduction.(Chapter 6)
3. AR AR-application with virtual alteration features to reduce display clutter when visualizing multiple robotic manipulation trajectories (Chapter 7)

The following are secondary contributions:

1. A novel AR-based robotic tool that enables robots to visually communicate their cognitive and sensory data (Chapter 4)
2. Open source repositories containing code for both the Augmented Reality applications as well as code used to run the robots used in this study,
3. Unity-based AR applications that can enable more people into the field of robotics, which potentially can increase workforce population by including older generations that are not so tech-savvy or even individuals with no programming or robotics experience.
4. Insights into applying a Detector-Response Test for Augmented Reality Head Mount Devices.
5. Insights for future designs of augmented/virtual/mixed reality objects in Human-robot Interaction scenarios.

6. A Unity template application for integrating augmented reality visuals for robotic reinforcement learning.

1.2.5 Overview

The rest of this dissertation is organized as follows:

Chapter 2 provides a summary of the related work in robotics, as well in augmented reality.

Chapter 3 describes the hardware and software that was used to perform this research.

Chapter 4 describes a study that investigates cases where users preference using an AR device to complete robot-related tasks.

Chapter 5 describes a study that investigates how AR can help people make better outcome predictions in the context of robotic fluid manipulations.

Chapter 6 describes a study that investigates how much information visualized is adequate enough for detecting failure in the context of object manipulation.

Chapter 7 describes a potential line for future work as well as a description of the initial first project, TRAinAR.

Chapter 8 provides a summary of this dissertation.

Chapter 2

Background & Related Work

In this chapter, we discuss literature that is relevant/related to the contributions of this dissertation. We begin with a review on visualization technology, Extended Reality (XR) which is an umbrella term that encompasses Augmented Reality (AR), Theory of mind and Robots. We then follow with a review on the state and challenges within the field of Human-Robot Interaction (HRI) in the context of communication, followed by a review on the use of AR within HRI, user acceptance of AR technology, and its impact on the Human Internal Physics Engine.

Visualization technology, rooted in the realms of psychology and cognitive science, embodies the convergence of human perception and digital innovation. At its core, this interdisciplinary field explores how individuals perceive, process, and understand visual information, leveraging insights from cognitive psychology to enhance user experiences and optimize information communication.

Central to the psychology of visualization technology is the concept of perception, which delves into how humans interpret and make sense of visual stimuli. Drawing from Gestalt principles, researchers in this field investigate how elements such as color, shape, and proximity influence the way information is perceived and organized. By understanding these perceptual mechanisms, designers can create visuals that effectively convey complex data while minimizing cognitive load [177]. Moreover, cognitive science plays a pivotal role in elucidating the mental processes underlying visual cognition. Through techniques like eye-tracking, researchers can

uncover the cognitive mechanisms involved in tasks such as pattern recognition, spatial reasoning, and decision-making. This knowledge informs the development of visualization techniques that align with human cognitive capabilities, facilitating more intuitive interaction and comprehension [91].

One prominent application of visualization technology within psychology is in the realm of data visualization. By translating abstract data into visual representations such as charts, graphs, and maps, researchers can facilitate data exploration and discovery. Through techniques like interactive visualization, individuals can engage with data in numerous ways, enabling deeper insights and more informed decision-making [83].

Furthermore, visualization technology intersects with cognitive science in the domain of extended reality, encompassing virtual and augmented reality (VR/AR). These immersive technologies leverage principles of spatial cognition and embodiment to create lifelike simulations and experiences. By manipulating perceptual cues such as depth, scale, and motion, VR/AR systems can induce powerful illusions that blur the lines between physical and virtual reality, offering new avenues for research and application in fields ranging from education, medical, manufacturing, and robotics [153].

2.1 Extended Reality (XR)

XR typically refers to *Extended Reality*, an umbrella term encompassing virtual reality (VR), augmented reality (AR), and mixed reality (MR). These technologies blend the physical and digital worlds, offering immersive experiences that can range from fully virtual environments to augmentations of the real world with digital elements. VR typically places users in entirely within virtual environments (full immersion), while AR overlays digital content onto the real world (partial immersion), while MR combines aspects of both by allowing virtual objects to interact with the real environment. For this dissertation, only AR is discussed and utilized.

AR technology is a system that combines virtual objects with the real world

in real-time, enhancing the user's perception and interaction with their environment with applications across domains such as entertainment, maintenance, training, and medical procedures [9]. Early work in AR laid the foundation for subsequent research endeavors. Sutherland introduced the a visionary concept of an immersive virtual reality (VR) system,"The Ultimate Display" [159]. In this work, Sutherland proposed an idealized computer-generated display that would seamlessly integrate with a user's senses, providing an immersive experience indistinguishable from reality leading to the concept of a head-mounted display (HMD) and outlined the vision for a virtual world. Mann's "Wearable Wireless Webcam" further demonstrated the potential for AR-enhanced vision capturing and augmenting the user's visual field in real-time [108].

With the development of AR hardware, particularly smartphones equipped with cameras, GPS, and accelerometers, access to AR experiences has increased [16], with applications spanning diverse domains, from gaming to education and healthcare, and robotics. Research in the 1990s witnessed significant advancements in tracking and visualization techniques. Azuma et al. introduced the concept of "ARToolKit," a software library enabling robust marker-based tracking for AR applications [7]. By accurately registering virtual content with physical markers, ARToolKit facilitated the creation of interactive AR experiences across various domains.

In healthcare, AR has gained attention for its applications in medical training, patient education, and surgical navigation. Studies have demonstrated the effectiveness of AR-enabled simulators in enhancing medical education by providing immersive, hands-on learning experiences [169]. Moreover, AR systems like the Augmedics xvision Spine System and Medivis have shown promise in improving surgical precision and patient outcomes by overlaying 3D reconstructions of patient anatomy onto the surgeon's field of view [55, 114]. Retail and marketing represent another domain for AR. Research by [41] illustrates how AR can enhance the shopping experience by allowing customers to visualize products in their physical environment before making a purchase decision. In the education sector, AR is

transforming traditional teaching methods by providing immersive and interactive learning experiences. Hsiao et al. evaluated the usefulness of augmented reality (AR) in teaching children [86].

2.2 Theory of Mind and Robots

The *theory of mind* is a fundamental concept in cognitive psychology and developmental psychology that gives individuals the ability to attribute mental states—such as beliefs, desires, intentions, and emotions—to themselves and others, and to understand that these mental states may differ from their own. This ability plays a crucial role in social interaction, empathy, and understanding others' behavior. It has been extensively studied in both human and animal cognition, with research suggesting that theory of mind development occurs gradually over childhood and is influenced by various factors including socialization, language development, and cultural context [134, 180, 22].

Humans also possess an internal physics engine—a cognitive mechanism that enables them to understand and predict the physical interactions between objects in the environment. This internal physics engine allows humans to intuitively grasp concepts such as gravity, momentum, and collision dynamics, enabling them to navigate the physical world, interact with objects, and anticipate the consequences of their actions [154].

While theory of mind and the internal physics engine represent distinct cognitive processes, there are interconnected aspects between them. For instance, in social interactions, humans often rely on their understanding of physics principles to infer the intentions and mental states of others. Observing how objects move and interact with each other can provide valuable information about the goals and intentions of agents manipulating those objects [65]

Moreover, the ability to attribute mental states to others and predict their behavior can also influence how humans perceive and interpret physical events. For example, when observing an agent interacting with objects in the environment,

humans may infer the agent's intentions and goals based on their understanding of physics principles and their theory of mind capabilities [12].

In the realm of robotics, the concept of theory of mind plays a pivotal role in the development of socially intelligent machines capable of interacting with humans in natural and intuitive ways. Incorporating theory of mind into robotic systems involves endowing them with the ability to perceive and infer the mental states of humans and other agents, including beliefs, intentions, desires, and emotions. Such capabilities can support effective communication, collaboration, and cooperation between humans and robots.

2.2.1 Human Internal Physics Engine

Humans have the ability to accurately predict the dynamics of physical objects has long fascinated researchers across various disciplines. This capability, often referred to as the "internal physics engine," [13, 100] enables individuals to anticipate the behavior of objects in their environment, facilitating tasks ranging from catching a ball to navigating through crowded spaces. While the exact mechanisms underlying this skill remain a subject of debate, numerous studies have shed light on the cognitive processes involved.

Early investigations by Gibson [66] emphasized the role of perceptual information in guiding predictive judgments about object motion. Gibson's ecological approach proposed that humans extract invariant information from their surroundings, allowing them to anticipate future events based on past experiences. Building upon this framework, recent neuroscientific studies have explored the neural substrates underlying predictive processing in the brain. For instance, work by [99] proposed the existence of internal models within the cerebellum, which simulate the consequences of motor commands and sensory feedback, thereby enabling accurate predictions of object trajectories.

2.2.1.1 Cognitive Prediction

The cognitive mechanisms underlying humans' predictive abilities have also been investigated extensively in cognitive psychology and computational neuroscience. One prominent theory, the "mental simulation hypothesis," suggests that individuals mentally simulate the dynamics of physical events using internal representations of physical laws [84]. According to this view, humans rely on mental imagery and analogical reasoning to anticipate the outcomes of interactions between objects, even in the absence of direct sensory input.

Moreover, research in developmental psychology has highlighted the role of experience and learning in refining predictive skills from infancy. Infants as young as three months old demonstrate rudimentary expectations about object motion, which become increasingly sophisticated over development [10]. These findings suggest that predictive abilities are not innate but instead emerge through interaction with the environment, underscoring the importance of sensorimotor experience in shaping the internal physics engine.

2.3 Human-Robot Interaction: Nonverbal Communication

HRI a multidisciplinary field that focuses on the study, design, and evaluation of robots and robotic systems that interact with humans. It combines fields for example from robotics, psychology, cognitive science, and human-computer interaction to understand and improve the ways in which humans and robots can collaborate, communicate, and coexist. HRI has seen significant advancements in recent years, with robots increasingly integrated into various domains, including healthcare, manufacturing, and household assistance [149].

Facilitating seamless collaboration between robots and humans is a primary objective in robotics [96]. It goes beyond basic interaction in human environments to include effective communication and cooperation [6, 42, 57]. As autonomous robots

increasingly join human workplaces, understanding communication challenges at different levels of interaction is crucial. Traditionally, commercial robots, lacking humanoid features, originated in industrial settings [85]. Previous HRI studies mainly focused on safety, operation, and control in regulated environments [69, 167]. However, as robots diversify, adapting to human expectations becomes key [102, 128].Non-humanoid robots, with their mechanical appearance, evoke different human responses [68]. Yet, as autonomous entities, accurately conveying capabilities in understandable ways is vital.Limited communication options for non-humanoid robots pose a significant hurdle, restricting the range of signals they can convey [15, 79].

These nonverbal communication techniques include gestures, facial expressions, body posture, and proxemic and are instrumental in conveying social cues, intentions, and emotions in HRI scenarios [17]. Gestures serve as a primary means of nonverbal communication in HRI, enabling humans to convey instructions, express emotions, and establish rapport with robots. Research by [119] highlights the importance of gesture-based communication in enhancing user engagement and comprehension of robot behaviors. By employing intuitive and culturally appropriate gestures, robots can effectively convey information and elicit desired responses from human users. Facial expressions constitute another essential component of nonverbal communication in HRI, enabling robots to express emotions and respond empathetically to human cues. Studies have shown that humanoid robots equipped with expressive facial features are perceived as more approachable and trustworthy by human users [92]. Furthermore, facial expression recognition algorithms enable robots to interpret human emotions and adjust their behaviors accordingly, fostering more natural and empathetic interactions [18]. Body posture and proxemics play a significant role in regulating social distance and establishing social norms in HRI contexts. Research by [56] emphasizes the importance of proxemic behavior in shaping human perceptions of robot autonomy and social presence.

2.3.1 AR for HRI

While robots hold immense potential to support human society across critical industries such as healthcare, manufacturing, and space exploration, their utilization has been constrained by challenges in control, supervision, and collaboration. A significant hindrance stems from the bidirectional challenge of human-robot communication, where robots often appear incomprehensible, and humans struggle to predict their capabilities or intentions. This communication gap leads to poor information exchange, impeding effective collaboration. To address these issues, researchers in human-robot interaction (HRI) are exploring novel methods of communication beyond traditional verbal or non-verbal cues. Visual communication, in particular, has shown promise in enhancing HRI, with advancements ranging from enhanced graphical displays to immersive virtual, augmented, and mixed reality (VAM) technologies. These VAM technologies offer a new medium for information exchange, allowing users to interact with 3D virtual imagery in their environment. Milgram's reality-virtuality continuum delineates the spectrum between reality and virtual reality, with mixed reality bridging the gap by combining real and synthetic imagery. The rise of VAM-HRI research demonstrates the potential of virtual reality (VR), augmented reality (AR), and mixed reality (MR) across various application domains, including education, social support, and task-based collaboration. AR visualization techniques, in particular, show promise in HRI by effortlessly communicating robot intent and behavior, such as visualizing navigational state and sensor data. These advancements offer new tools for robot design, experimentation, and programming, contributing to the advancement of human-robot collaboration and interaction.

Chapter 3

Systems Description

This chapter provides an overview of the hardware and software that was used to conduct the research described in this dissertation.

3.1 Hardware

3.1.1 Turtlebot2

The TurtleBot 2 is a popular and versatile mobile robot platform designed for education, research, and prototyping in the field of robotics. Developed by Willow Garage and now maintained by Open Robotics, the TurtleBot 2 combines hardware and software components to create a customizable and accessible robotic platform. Its base platform features two motorized wheels and a caster wheel for stability, enabling locomotion in indoor environments. Equipped with sensors like the Astra Depth Camera and RPlidar, it offers perception capabilities such as depth sensing and RGB imaging for tasks like 3D mapping and object detection. Integrated with an onboard computer, typically a laptop, the TurtleBot 2 runs the Robot Operating System (ROS), serving as the central processing unit for executing software algorithms, processing sensor data, and facilitating communication with external devices. (See Figure 3.1)

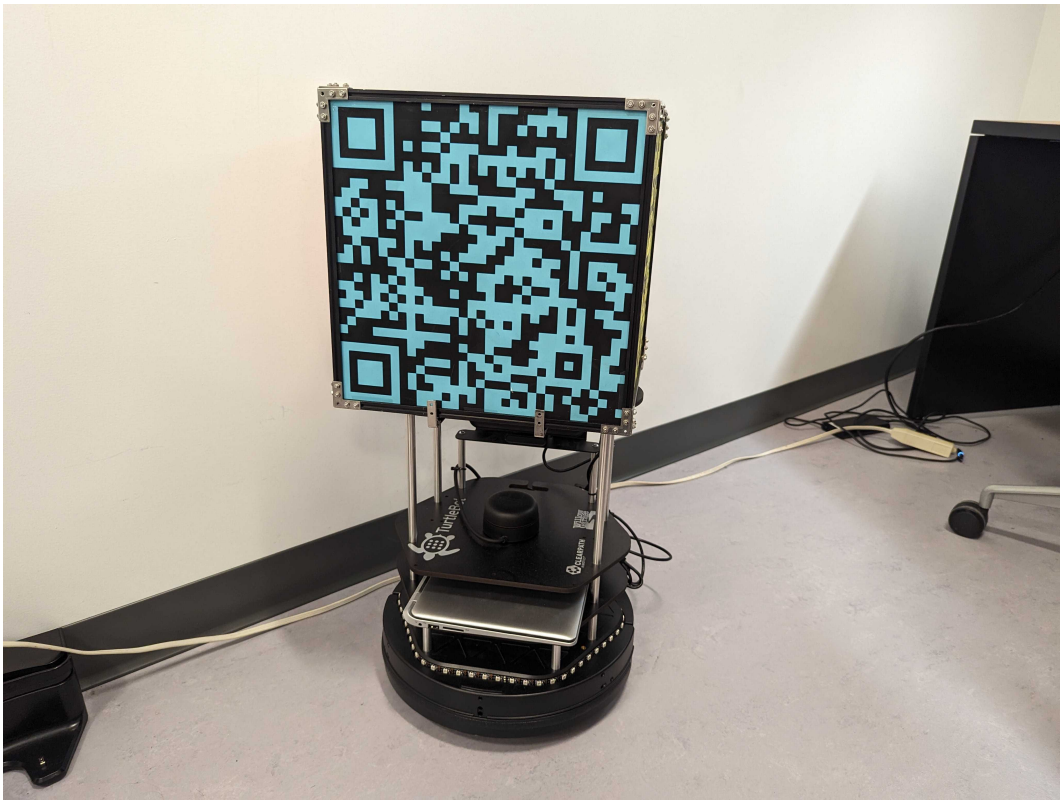


Figure 3.1: Image of the Turtlebot2 robot used in this dissertation. The QR-printed cube attached to the top of the robot serves a target image for tracking purposed discussed further in 3.2.3

3.1.1.1 Sensors

The Turtlebot2 is equipped with the Astra depth camera is a 3D depth sensing device developed by Orbbec Technology and is designed for a wide range of applications including robotics, mixed reality, 3D scanning, and gesture recognition. The Astra camera was mounted to the Turtlebot's middle compartment platform facing the robot's heading that captured 640 x 480 RGB images as well as a 3D point cloud. In addition, a RPLidar is a popular 2D laser range scanner (LIDAR) developed by RoboPeak. It is widely used in robotics, autonomous navigation, and mapping. The RPLidar sensor is typically fixed to either the top or bottom platform of the Turtlebot2 and is connected via USB connection to the laptop that used to control the robot. (See Figure 3.2 for an image of the Astra camera and RPLiadar mounted to the Turtlebot2)

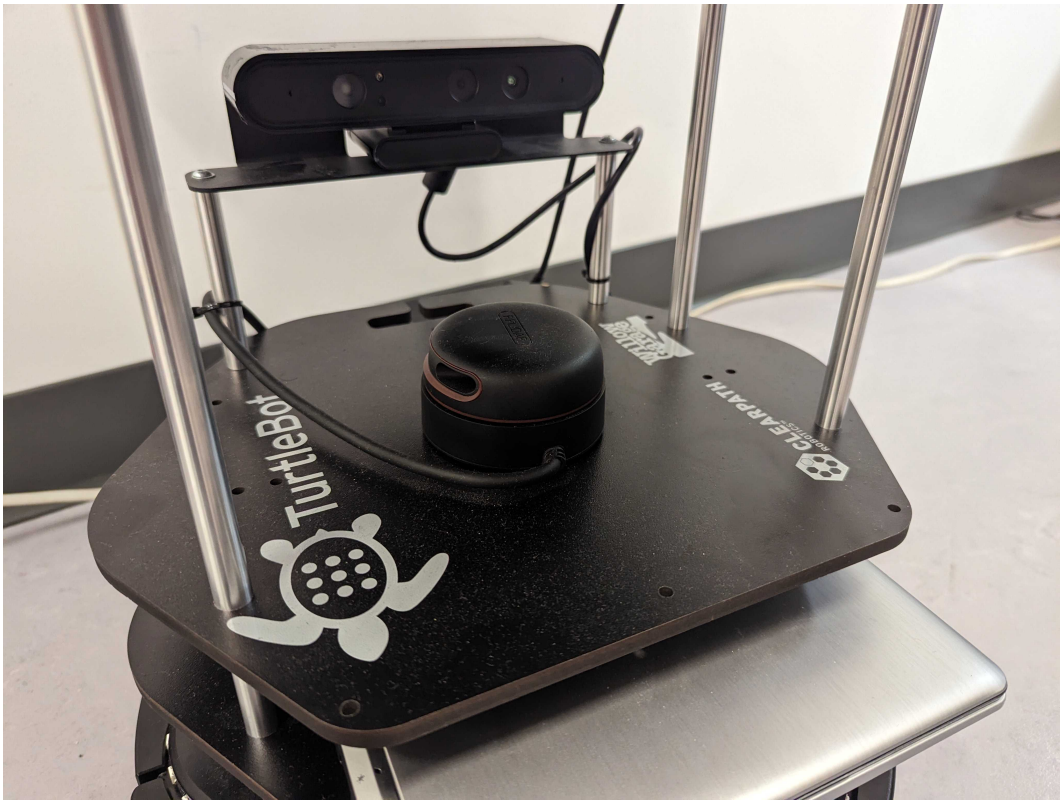


Figure 3.2: Image of the RPLIDAR sensor and Astra camera mounted to the Turtlebot2 that were used in this dissertation.

3.1.2 Kinova Gen3 Lite

The Kinova Gen3 Lite robot is a versatile and lightweight robotic arm with 6 degrees of freedom and a two finger gripper designed for a wide range of applications in research, education, and industry. The Kinova is compatible with Robot Operating Software (ROS)

3.1.3 Microsoft Hololens2

The Microsoft HoloLens 2 (see Figure 3.4) is an advanced mixed reality (MR) headset that builds upon the foundation of its predecessor, the HoloLens. It offers an immersive computing experience by blending virtual content with the real world. We utilized the Hololens2 for two of our studies discussed in Chapter 5 and Chapter 6.

Here's a detailed breakdown of its features:



Figure 3.3: Image of the Kinova Gen3 Lite robot used in this dissertation. Towards the base of the arm is a cylindrical target image used or tracking the robots pose. See section 3.2.3) for further details.

Optics and Display: The HoloLens 2 features a custom-built holographic processing unit (HPU) and an improved display system. It employs waveguide optics to project holograms directly onto the user's field of view. The display has a wider field of view compared to the original HoloLens, providing a more immersive experience.

Comfort and Ergonomics: Microsoft has prioritized comfort with the HoloLens 2. It has a more balanced design, with the weight distributed evenly around the head. The device is adjustable to accommodate various head sizes and can be worn comfortably for extended periods.

Gesture Recognition and Interaction: One of the key features of the HoloLens 2 is its enhanced gesture recognition capabilities. It uses hand tracking technology to accurately detect hand movements and gestures, allowing users to interact with holographic content more intuitively. This includes gestures such as



Figure 3.4: Image of the Microsoft Hololens2 used in this dissertation.

pinch, grab, and drag.

Eye Tracking: The HoloLens 2 integrates eye-tracking technology, enabling more natural interactions and enhanced user experiences. Eye tracking can be used for example to provide information on objects that a user may be focused on such as small window that pop ups after gazing on an object for a specific time period.

Spatial Mapping and Environment Understanding: The device is equipped with sensors and cameras that enable it to map and understand the user's environment in real-time. This allows virtual content to interact seamlessly with the physical world and enables applications such as spatial anchoring, where virtual objects can be placed and persistently anchored in the user's environment.

Mixed Reality Applications: The HoloLens 2 is designed to support a wide range of applications across industries such as manufacturing, healthcare, education, and entertainment. These applications leverage the device's capabilities to create immersive training simulations, interactive educational experiences, remote

assistance solutions, and more.

Connectivity and Integration: The HoloLens 2 is equipped with built-in Wi-Fi and Bluetooth connectivity, allowing it to connect to other devices and peripherals. It also integrates with Microsoft's Azure cloud platform, enabling seamless access to cloud-based services and data.

Development Tools: Microsoft provides a comprehensive set of development tools and resources for building applications for the HoloLens 2. This includes the Mixed Reality Toolkit (MRTK), which offers libraries, APIs, and tools to streamline the development process.

Overall, the Microsoft HoloLens 2 represents a significant advancement in mixed reality technology, offering improved comfort, enhanced interactions, and expanded possibilities for immersive computing experiences.

A more detailed guide of the Microsoft Hololens2 can be found on their website.

3.1.4 Samsung Galaxy Tab s8 Tablet

The Samsung tablet is a premium Android tablet that supports AR development. Gaming engines such as *Unity* can build and deploy AR applications directly to the tablet.

3.1.5 Mechanical button for Detector-Response Test

The mechanical button is a handheld device that uses an Arduino Nano*, a microcontroller, enclosed by a 3D printed box and a plastic red circular extrusion that is designed to be easily pressed by a finger or thumb that communicates with a laptop via USB connector that registers a button press by sending a signal back to the computer.

*https://store.arduino.cc/products/arduino-nano



Figure 3.5: Image of the mechanical button used in Chapter 6 for the Detector-Response Test.

3.2 Software

3.2.1 Robot Operating System (ROS)

ROS is an open-source robotics middleware framework that provides a collection of software libraries and tools to help developers build robot applications. Developed by Willow Garage and now maintained by the Open Robotics organization, ROS has become the industry standard for robotics software development due to its flexibility, scalability, and extensive community support. Specifically, ROS versions Kinetic and Melodic were used to control the robots featured in this dissertation. Further information about ROS can be found on their website.

3.2.2 Unity

Unity is a cross-platform game engine developed by Unity Technologies that is widely used for creating 2D, 3D, AR, and virtual reality (VR) applications across various platforms, including desktop, mobile, console, and web. Additional information can be found on their website.



Figure 3.6: Image of the Tablet System used in this dissertation. (Top Left): Mounting arm, (Top Right): Samsung Galaxy Tab s8, (Bottom): Tablet mounted to table

Specifically, versions of 2019 and 2020 were exclusively used in this dissertation to develop and deploy to Mixed Reality devices such as Augmented Reality (AR) supported smartphones, tablets, and the Microsoft Hololens2.

3.2.3 Vuforia

Vuforia is an AR software development platform owned by PTC (formerly Qualcomm). It enables developers to create immersive AR experiences for various industries and applications. Vuforia offers a comprehensive suite of tools for creating augmented reality (AR) content, including image recognition, object recognition and tracking, and environmental understanding. Developers can leverage Vuforia's advanced computer vision algorithms to recognize and track images (acting as fudicial markers), objects, and environments in the real world, enabling AR experiences to overlay digital content on specific images or patterns, interact with physical objects, and adapt to the user's surroundings. This includes the ability to define

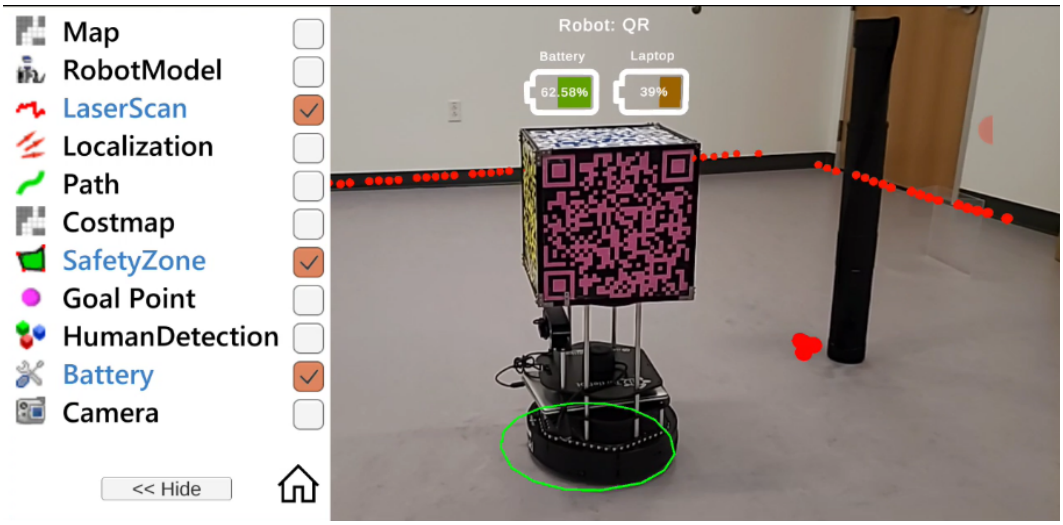


Figure 3.7: Image of a screenshot on a Samsung Galaxy Tab s8 running a Unity-based AR application to visualize robot data. The left panel includes a list of robot data visualization options that a user can toggle, while the right shows the robot and the corresponding robot data visualized.

targets for recognition and tracking, such as images, objects, or spatial locations, and anchor virtual objects onto real-world surfaces for seamless integration into the environment.

In this dissertation, vuforia's target image recognition was primarily used to track our robots. Developers can create and manage a database of target images using the Vuforia Target Manager, a web-based tool provided by Vuforia. In the Target Manager, developers can upload images or predefined markers that they want their AR application to recognize. These images can be photographs, logos, symbols, posters, or any other visual content. Vuforia supports different recognition modes to suit different application requirements which include Single Image and Multi-Target.

Here, a feature rich image can be fixed to a robot to track the robot's position and orientation. Once the robot's location in space is known, AR visuals that are parented with the robot can be rendered by the AR device in their correct locations. Figure 3.8 is from the SENSAR project described more in Chapter 8.

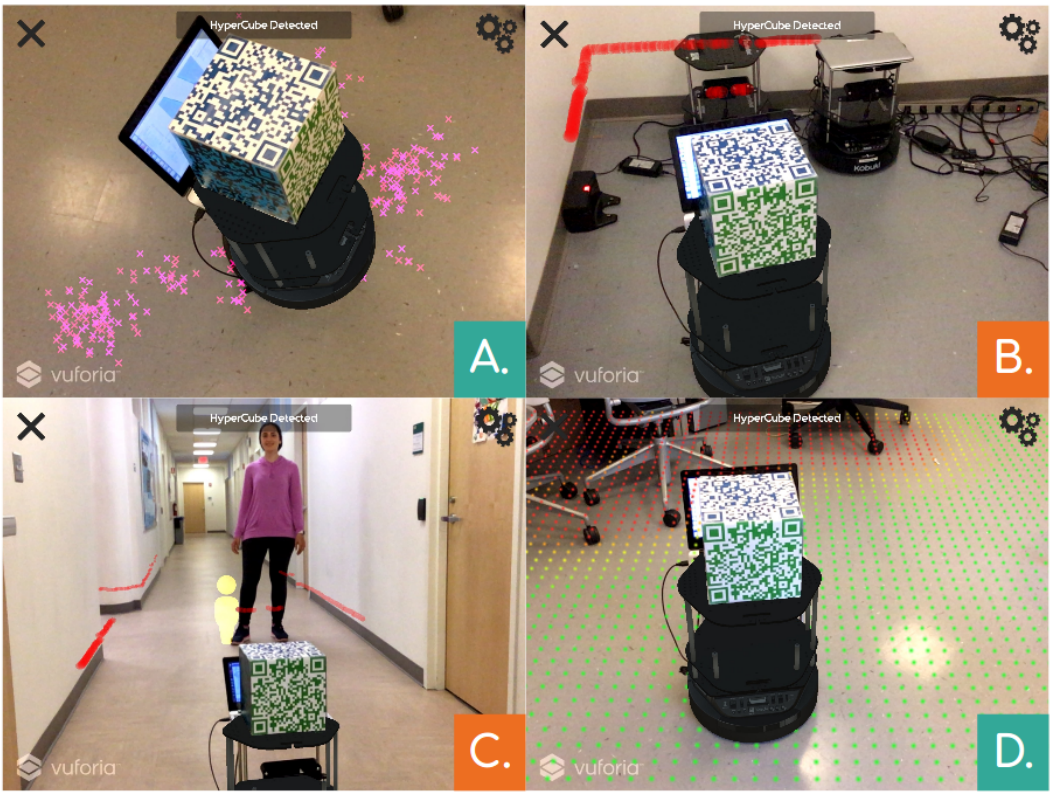


Figure 3.8: Image of a robot with a Multi-Target image tracker fixed on top of a Turtlebot2 robot for tracking. Once tracked, AR visuals programmed in Unity can be rendered by the AR device. This figure shows various types of visual data you can display with in AR. (A) Localization, (B) Laser scan, (C) People Detection, (D) Cost Map.

3.2.4 Database Server for Data Collection

A web interface loaded on the Samsung Galaxy Tab S8. This interface, programmed in Python using Flask[†], was responsible for collecting user data including objective and subjective measures.The web interface populated a MySQL[‡] database with collected data. The database was hosted in a Docker[§] alongside a container running a phpMyAdmin[¶], which provided an administration interface that was used for both debugging and creating XML backups of the database. Additionally, a cronjob was written to sync the latest SQL files to an onsite backup server to provide redundant

[†]https://flask.palletsprojects.com/en/3.0.x/

[‡]https://www.mysql.com/

[§]https://www.docker.com

[¶]https://www.phpmyadmin.net/

data storage. To ensure that the database could be rolled back if necessary, the onsite backup server copied the SQL files to timestamped backup directories every 6 hours.

3.3 Software: SENSAR[§§]

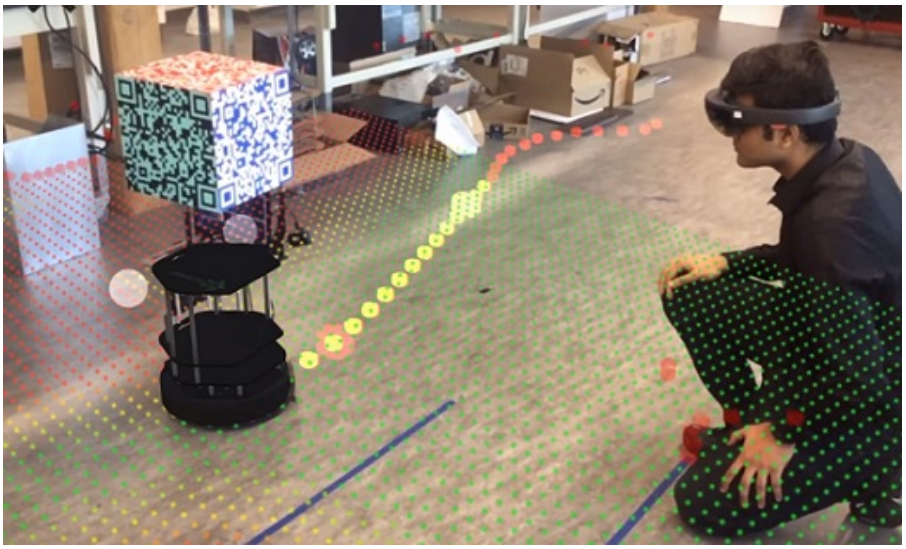


Figure 3.9: Interactive virtual space shared by a human and a mobile robot. The Hololens renders the robot's Laser scan data (red markers on human), OccupancyGrid data (red/green dots in grid formation), and planned path trajectory (red/yellow markers).

3.3.1 Introduction

For intelligent robots to operate in the same environment as humans, they must communicate their intentions, behaviors, and knowledge in a way that is interpretable by humans. Robots rely on sensors and algorithms to support functions such as localization, obstacle avoidance, and object manipulation; however, robots can misidentify objects and communicate their misperceptions of the world, which

[§§]**This chapter is based on the following paper:** Andre Cleaver, Faizan Muhammad, Amel Hassan, Elaine Short, and Jivko Sinapov. SENSAR: A shared reality with intelligent robots for collaborative human-robot interaction. In AAAI Fall Symposium Series: Trust Explainability in Artificial Intelligence for Human-Robot Interaction (AI-HRI), 2020

can lead to undesired outcomes. As a result, humans may not understand what a robot is trying to do, with adverse effects on human-robot collaboration. One way to solve this problem is to allow the human collaborator to directly see a robot's perception to ensure the robot is performing adequately [168].

Augmented Reality (AR) renders spatially-grounded images over the real world. By leveraging AR, robot data that would typically be "hidden" can be visualized by rendering graphic images over the real world using AR-supported devices (e.g., smartphones, tablets, or the Microsoft HoloLens) [58]. For example, a robot's depth sensor data rendered directly over a workstation shows not only what the robot "sees" but also what the robot does *not* "see": that is, it allows the human to see what the robot could not detect. AR gives humans the advantage of placing the robot's sensory data in the context of the real world unlike traditional visual programs (e.g., rviz[‖]).

SENSAR (Seeing Everything iN Sitiu with Augmented Reality) is an AR-mediated communication tool. We describe the design of our system, which uses ROS and Unity, along with the data processing routines to convert raw data to visuals. Our system can create an interactive reality space containing the robot's sensory data and cognitive output that a user can access to reach a "common ground" with the robot (see Figure 3.10). In a technical demonstration of the system, we show that visualizing sensors and intended actions can be beneficial for humans in areas such as debugging, education, and serve as a tool for HRI studies.

3.3.2 Related Work

Currently, there are frameworks that already exist that display robot data using external lights and projections [54, 24]. Although these methods have the advantage of not requiring the user to hold or wear a separate device, they are not without limitations. These solutions require modifying the robot with hardware and staging the environment with flat surfaces to view the projections. Occlusions are another inevitable problem with this approach if any objects or humans themselves are

[‖] http://wiki.ros.org/rviz

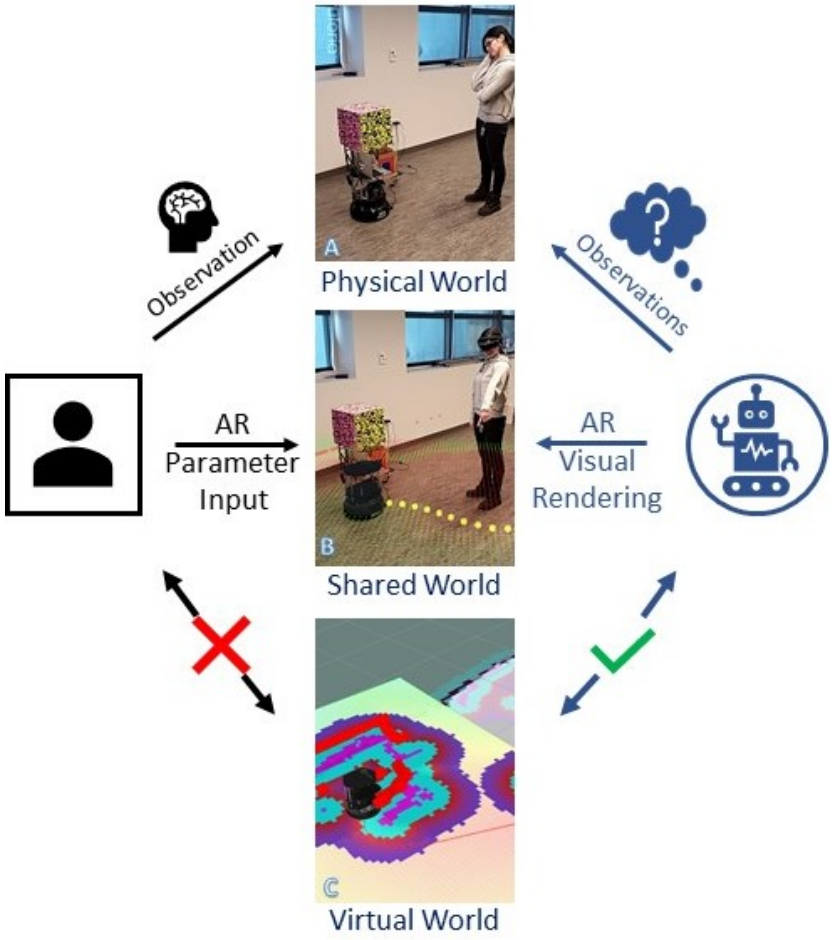


Figure 3.10: (Right) Physical world known by a human; robots are unaware of most objects within environment, (Left) Virtual World accessed only by the robot what includes the robot's algorithms and sensor readings, (Center) a shared reality world accessed by both humans and robots. A robot renders its path plan using markers which is seen by human.

positioned between the display surface and the projection source. AR does not face this problem as graphical images are rendered over the real-world, thus eliminating the need for any changes to the existing robot or environment.

AR has been integrated with several robotic platforms with a focus in areas such as navigation, programming, and education [173, 63, 30]. Work by [30] explored AR as an education tool for K-12 students. Middle school students were observed debugging their assigned robots (EV3 Kit) through Tablets and initiated group discussions around sensor readings. The EV3 kit contained actuators and sensors and such as touch, color detection, sonar, and gyro. Although their system was

successful in visualizing the robot's data, the system can only function with the EV3 robot. We demonstrate SENSAR with a Turtlebot2 robot, a popular robotic platform for research; however, any ROS-based robot can work with the SENSAR system. In addition, SENSAR is highly modular as multiple sensors including the type of sensors provided in the EV3 kit can be connected to the robot such as depth and Lidar sensors. SENSAR is a continuation of prior work by [117].

3.3.3 SYSTEM ARCHITECTURE AND DESIGN

SENSAR is an AR system** enabling users to "see into the mind" of a robot. Data collected by an intelligent robot is processed through various filtering techniques (e.g., downsampling) to render the output in a visual form that is intuitive for a user (See Figure 3.11). The functional requirements of SENSAR included:

- Customizable for different sensors and common data structures within a robotics system
- Operational without external tracking or cloud service
- Versatile to multiple AR-supported devices

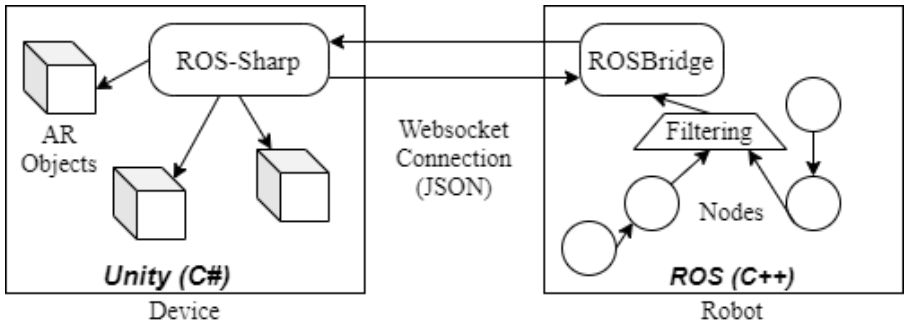


Figure 3.11: System Architecture - Our system communicates various data types between the Robot and AR device through a web-socket connection. Sensor data collected through nodes go through a filtering layer for down-sampling and to remove non-numerical data points (i.e. NaN).

**The SENSAR package repositories can be found at: https://github.com/tufts-ai-robotics-group/arfuros_ros.git and https://github.com/tufts-ai-robotics-group/arfuros.git

3.3.3.1 Hardware and Software

SENSAR requires two components: an intelligent robot and an AR device. A *Turtlebot2* served as the intelligent robot; it is controlled through the Robot Operating System (ROS) [136] running on Ubuntu 16.04. The robot is equipped with an *RPLIDAR A2M8 Laser Scanner* and an *Astra Camera* depth sensor. The robotic sensors and cognitive data are extracted and filtered from ROS using Python and C++ scripts. For the AR device, we used a Microsoft Hololens, a Samsung Galaxy S9 Android smartphone device with a 12 MP Camera, and an iPad to render visuals over the real-world. Our system can be installed on any smartphone, iPhone, or tablet device that meets the hardware requirements to support AR applications.

The AR device utilizes *Unity*[††] to create visualizations to project data around the robot. *Vuforia*[‡‡] is used to establish physical tracking of the robot using a target image. Once the pose of the robot within the incoming video feed is detected, the data received from the robot can then be projected onto the camera feed. In ROS, *ROSBridge* provides a JSON interface to the topics native to ROS, whereas in Unity, *ROS Sharp* provides C# functions and classes to interact with *ROSBridge* through JSON. Through a WebSocket connection, ROS messages of any type from the robot come out as native C# data structures at the other end of the pipeline.

3.3.3.2 Visualization Options

Below is a list of current visualizations that are equipped with the SENSAR system. SENSAR is an ongoing project and we plan to add additional visualizations in the future.

Localization Particles - This data type is based on the Monte Carlo localization algorithm or *Particle Filter Localization* that robots use to localize themselves within a given map [113]. Figure 3.12A show markers in pink visualizing the robot's belief about its location within a given map.

[††] https://unity.com/

[‡‡] https://developer.vuforia.com/

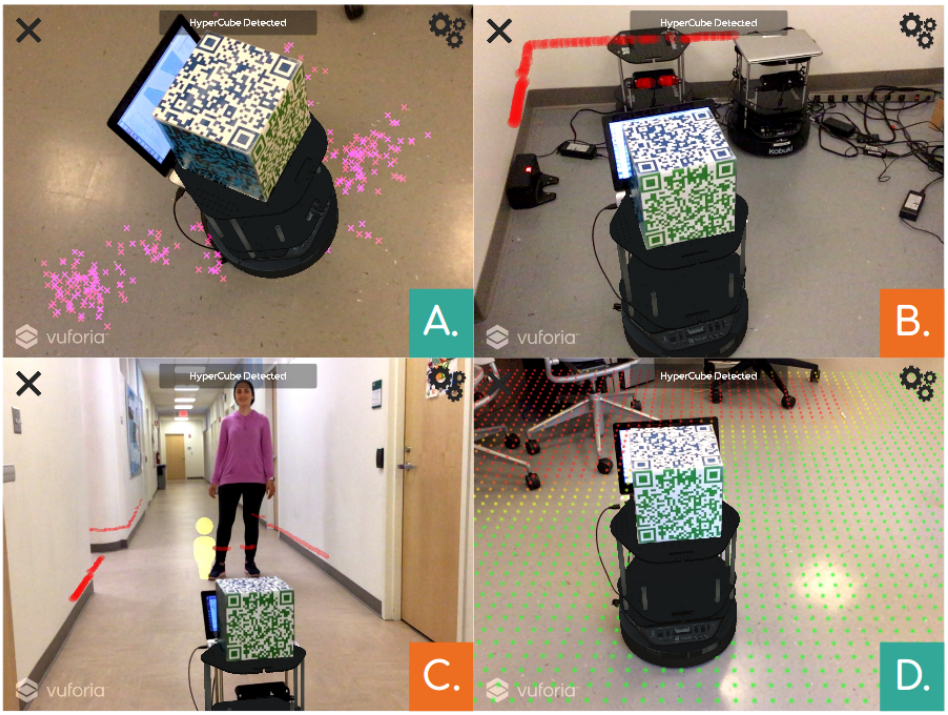


Figure 3.12: This figure shows the first design of the robot and the types of visual data you display with AR. (A) Localization, (B) Laser scan, (C) People Detection, (D) Cost Map.

Laser Scan - A 2D LIDAR reading is shown in figure 3.12B Here, the laser scan detects the walls and objects with red markers.

Human Detection - A human detection algorithm leverages the Laser Scan data as input and compares the measurements against the profile of a pair of human legs. Legs are used as a target feature due to the placement of the LIDAR sensor. A human avatar is rendered in place of a coordinate data point that represents a detected human. Figure 3.12C shows the Human Detection visual as seem by the yellow human avatar.

Occupancy Grid (Cost Map) - Cost Map is a measure of occupancy within the surrounding area of a robot. The robot reports a planar grid consisting of cells using sensory data and information from a static map. Each cell represents a probability of occupancy [0-1]. The information for each cell influences the robot's path trajectory by selecting cells with low probability of containing

an obstacle. For visualization, we related the probability value of a cell to correspond to a color. Figure 3.12D shows the *Cost Map*. Here, the robot detects the chairs in its surroundings and then renders red markers (high probability of occupancy) in their location and green markers for areas that are clear.

Path Trajectory - Path Trajectory visual represents the robot's future trajectory to a selected destination point as a series of markers on the ground plane. The trajectory is generated by the robot's internal path planner algorithm. Figure 3.13 Top shows the robot's path trajectory around the obstacle (green stuffed animal).

Turning Signal - The Turning Signal data type alerts a user on the robot's intended direction of navigation by flashing an arrow towards the respective direction. Turn Signal mimics a motor vehicle's turning signal system. Figure 3.13 Bottom shows the robot's flashing an arrow towards the direction it will move.

Depth Camera Sensor (*PointCloud)* - PointCloud renders the depth sensor data as particles in 3D space. Filtering and segmentation techniques are applied for model detection algorithms. We expanded on visualizing PointClouds with the development of *RAIN* discussed in section 3.4.

3.4 Software: RAIN[§§]

Robots rely on noisy sensors to accomplish their tasks. For example, depth camera sensors output 3D data which are then interpreted by visual algorithms to segment objects of interest. These algorithms often have many parameters, each of which

[§§]**This chapter is based on the following paper:** Andre Cleaver, Elaine Short, and Jivko Sinapov. RAIN – A vision calibration tool using augmented reality. In ACM/IEEE Conference on Human-Robot Interaction (HRI), Workshop on Virtual, Augmented and Mixed Reality for Human-Robot Interaction (VAM-HRI), Cambridge, UK, 2020

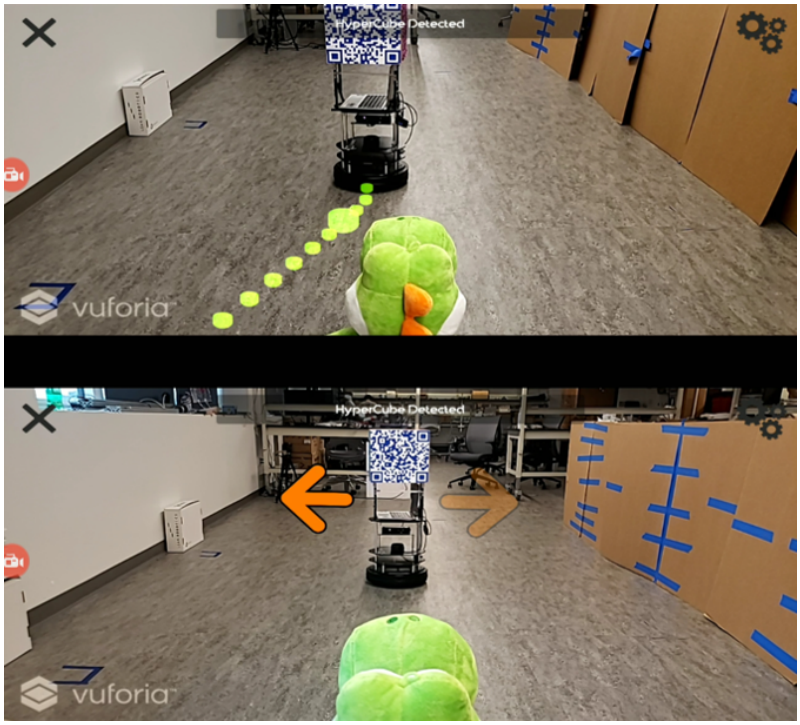


Figure 3.13: Visualization projected by the robot to indicate intended direction of motion around an obstacle. Top image is path trajectory using yellow markers. Bottom image is turn signals which operates similar to turn signals of a motor vehicle.

needs to be set appropriately for a given task and environment. The success of a robot's task depends on this calibration which is often performed with human expertise. Robots often need their sensor algorithms re-calibrated due to environmental changes, malfunctions, or periodic maintenance to continue operation. However, current approaches to parameter-setting and calibration require significant expertise, leaving a robot out of commission for periods of time unless a new, user-friendly approach to vision calibration is introduced.

In this paper, we propose and describe the development of an Augmented Reality (AR) system designed to allow non-expert users to calibrate a robot's visual perception software pipeline. AR has helped robots convey their motion intent as well as display their cognitive and sensory data onto the real-world giving humans an intuitive visual aid from the perspective of the robot [117, 172]. AR technology can create an interactive virtual space containing the robot's sensory data and cognitive output (see Figure 3.9) that a human can access to detect and address malfunctions.

We introduce a "shared reality" framework for understanding the role of AR in human-robot interaction (HRI) for vision calibration, describe the design of a system for using AR to calibrate a multi-parameter RGBD object segmentation pipeline, and describe a planned study to validate the described system.

Our proposed framework, RAIN (Robot-parameters Adjusted In No-time), allows the robot to share the state of its sensor data through an AR interface available to the human user. Our proposed system includes a novel approach to parameter-setting which allows naive users to appropriately calibrate complex visual perception pipelines. Using our approach, users can adjust the filtering/segmentation parameters through the AR device that directly modifies the computer vision pipeline running on the robot. The users can see the results of their modification over-layed onto the real world through the AR device, allowing them to choose appropriate parameters without necessarily understanding how the underlying pipeline works.

Some recent work on automatic sensor calibration has attempted to eliminate the need for a human in the loop entirely. For example, algorithms have been

developed that do depth segmentation for object grasping using supervised and reinforcement learning techniques [141, 103]. Other segmentation techniques have leveraged prior image processing techniques and integrated depth information [1]. However, in real-world environments that change over time, it will still be helpful (or even necessary) to have an end user validate that the final parameterization is correct. Our work provides a method for doing this, and relates to work in HRI on establishing common ground between humans and robots and to the literature on augmented reality for robotics. The contribution of this work is (1) identifying an intuitive visual representation of depth information and (2) the development of AR-mediated algorithmic calibration tool for all user experience levels.

3.4.0.1 Reaching Common Ground with a Robot

"Theory of Mind" is a set of social skills humans use to attribute beliefs, goals, and desires to other individuals. Within HRI, humans form mental models of robots that express social cues and gestures with the same attributions. Furthermore, the *Common-ground Theory* suggests that mutual understanding between humans during communication is due to coordination of shared information [32]. The common ground theory is then essential towards effective human-robot interactions as it can elicit finer mental models of a robot from a person [97]. Common ground in HRI is reached once the beliefs and perceptions of the physical world as observed by the robot match those observed by the human. Therefore, we focused on an interface design that allows users to visualize the robot's sensor data in the real world to establish a form of common ground and allow the user to set parameters for the vision pipeline.

Related works include common-ground frameworks for human-computer and human-robot interactions using verbal [127, 19, 25, 182] and non-verbal communications [50, 20]. However, this work has not been extended to human-robot interactions beyond natural language. A pilot study by Cheli *et al.* [30] explored AR as an education tool for K-12 students. Middle school students were observed debugging their assigned robots (EV3 Kit) through tablets and initiated group discussions

around sensor readings.

3.4.0.2 Augmented Reality for Robotics

AR enables the rendering of computer graphics on to the real-world in real-time [8, 72]. This differs from Virtual Reality (VR) that renders computer graphics over the entire physical environment which places a user into a fully immersive virtual world [111]. Thus, robots can leverage AR as a medium for communication and interaction in HRI. AR interfaces such as the Microsoft HoloLens have shown promise in further enhancing natural HRI by aligning the perspectives of humans and robots [143, 142]. For this study, relevant works include approaches that have robotic systems that communicate through an AR interface. Walker *et al.* [172] explored an AR design space that conveyed a UAV's motion intent through various explicit and implicit designs. Their study showed that a design that renders more visual explicit information on the environment resulted in better task efficiency and was rated as more clear, usable, and the robot seen as a better member of the team. Muhammad *et al.* [117] had developed an AR design communication tool that displays a robot's cognitive and sensory data as well as prompting for human intervention in order to create a shared reality between humans and robots for communicating and problem-solving. Gadre *et al.* [62] enabled users to create waypoints through an AR-interface to control the motion trajectory of a robotic arm.

Although these prior works have developed tools to visualize a robot's perception of the environment, no prior work has studied how AR can be used to improve perception-pipeline calibration by non-expert users. In this work, we describe a set of design possibilities for such AR-enabled calibration, and show that using our system can allow a non-expert user to correctly and quickly calibrate a perception pipeline.

We designed an AR mobile system to render processed robotic sensory data as visual particles with an interactive UI allowing users to modify the distribution of the particles, and we plan to deploy the system in a user study as described below:

3.4.0.3 Calibration Task

We consider the task of segmenting objects from a 3D point cloud. Common issues include mistaking two objects as one, losing objects within the background, or misidentifying an object due to a noisy measurement. We use the Point Cloud Library (PCL) [§§] as the segmentation pipeline, which contains a variety of filtering and segmentation algorithms for detecting objects in 3D data. The specific steps in the pipeline include a voxel grid filter, a pass-through filter, outlier removal and cylinder segmentation. Each of these components has a set of parameters (e.g., the size of each voxel, radius limits for cylinder detection, etc.).

3.4.0.4 System Design

In this section, we describe the AR system which allows users to visualize 3D point clouds while calibrating the robot's visual processing module. The system is composed of three main components:

Depth Camera Sensor: We used the Astra Camera sensor for gathering point-cloud data sets which are then processed by visual algorithms. This camera is commonly used with the Turtlebot2 robotic platform. The depth camera sensor was plugged into an ASUS laptop installed with Ubuntu 16.04 LTS and launched by Robot Operating System (ROS) [136].

AR Device: We used a Samsung Galaxy S9 Android smartphone device with a 12 MP Camera to run the application. It should be noted that the application that runs the proposed design space can be installed on any smartphone or tablet device. The application has the ability to render visual markers similar to those displayed in RViz, the default visualization software that is part of ROS.

Communication Framework: A customized framework ensured that the visualizations and interactive input fields were appropriately displayed. First, the visual prototypes described in 3.4.1.2 were developed in *Unity* [¶¶]. Displaying the vi-

[§§] http://pointclouds.org/

[¶¶] https://unity.com/

sual information accounting for the pose of the robot is handled through *Vuforia* [***], an augmented reality software development kit, which enables the rendering of computer graphics onto real-world environments. A websocket connection links the robot platform to the AR device. This link enables ROSBridge to send topic messages as JSON types within ROS to Unity. ROS-Sharp is responsible for the conversion of ROS topics to JSON messages. C# scripts that run within Unity handle the feedback responses.

3.4.1 Proposed Study

We plan to study the effect of several different visualization types on users' ease and effectiveness at calibrating the vision pipeline.



Figure 3.14: Uniform distribution of rendered particles that correspond to the filtered output of the raw data. (Left) Low density and (Right) High density.

3.4.1.1 Interaction Design

Preliminary validation of the system was conducted in a tabletop workspace with a cylinder detection task. The workstation consisted of a workbench and a depth camera sensor attached to an adjustable arm-bar secured to the workbench resulting in a front view of the workbench as seen in Figure 3.14. Various cylindrical objects are placed on the workbench which can be detected by the sensor. A user can access the visualizations by placing the target image within the AR devices field of view which will render the graphics on the screen. Users then can adjust the parameters displayed on the AR device. Figure 3.15 shows an adjustable radius parameter at the bottom right of the interface.

[***]https://developer.vuforia.com/

3.4.1.2 Visualization Designs

We consider four different visualization modes for displaying the 3D point cloud data set to the human user:

1. **Uniform density**: Raw point-cloud data points are rendered uniformly after voxel-grid filtering, as shown in Figure 3.14.

2. **Non-Uniform density**: This visualization identifies the ground plane and renders the plane with a sparse distribution of markers, while identified objects will be rendered with a dense surface area of markers.

3. **Heterogeneous density**: Identified objects are rendered with visual markers that differ from the ground plane. (i.e. slightly larger, white in color, and spherical in shape, etc.)

4. **Negative Space density**: Within this visualization, the particles are rendered around the space of detected objects.

3.4.2 Outcome Measures

Objective and subjective measurements will characterize the effectiveness of the calibration tool. Task efficiency measures the total time participants spent completing the task (lower times indicated better performance/efficiency). A participant's recorded time will only be included in the analysis if no technical issues occur. Technical issues include any freezing of the AR-Device or on the robot end. Consistent start and end times for each participant will ensure fair results.

A percentage of the objects detected will be recorded to measure accuracy. As for the subjective ratings, a 7-point scale Likert-style questionnaire common among HRI papers, will be utilized to measure user perceptions and preferences. Scales will rate *Design Clarity*, *Design Usability*, and *Recommendation for other users*. Open-ended responses will be recorded for any improvements or changes to the study and will be administered at the conclusion of the study. They will include

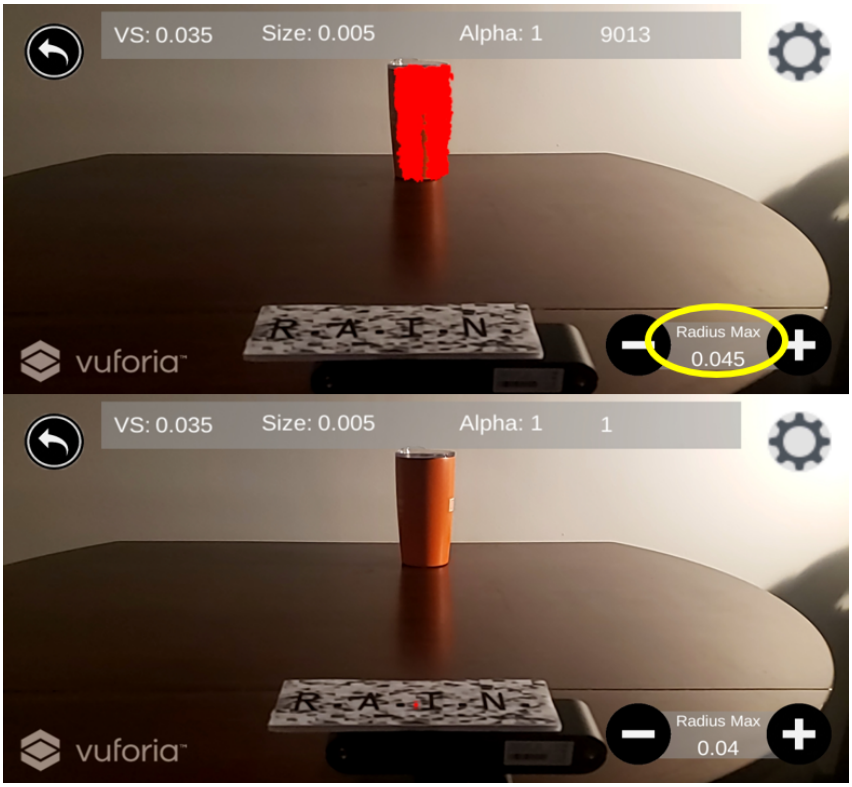


Figure 3.15: Red particles rendered over a detected real-world object. The particles correspond to points in the raw data that fit a cylinder model of a given radius. A user adjusts the radius parameter to detect the object.

statements that refer to "working with the AR interface", "difficulty to correct the robot perception", "AR-device provided enough information". We will then analyze the data using a one-way Analysis of Variance (ANOVA) with experimental condition as the fixed effect. Post-hoc tests will evaluate the design methods against a baseline condition. In addition, the NASA-TLX workload survey metric will measure cognitive workload when analyzing the visualization techniques [155, 80].

3.4.3 Procedure

(1) First, participants are read identical instructions for the study and the task they will perform, and a consent form is then obtained. In addition, the participant is randomly selected a visualization (i.e. AR Visualization Designs) they will utilize. (2) Next, participants familiarize themselves with the AR-device and the functionality of the application; this will take approximately 5 minutes. Participants who

are assigned to the baseline condition are verbally instructed that the robot will make observations in the world and can be seen through a non AR-device. (3) Participants will then perform the instructed task within 10 minutes. Specifically, participants will try to identify all cylindrical objects placed on a tabletop by only using the interactive buttons on the AR interface. (4) After the allotted time has elapsed, participants will be notified that the task is over, and a post-questionnaire will be administered.

3.4.4 APPLICATIONS

3.4.4.1 Conveying Navigational Intent

We are exploring how SENSAR can leverage the Path Trajectory and Turning Signal data types for users to interpret the robot's motion intent. These navigation visuals will be evaluated using videos clips. Participants will watch segments of the robot approaching an obstacle. The robot will travel either to the left or right around an obstacle as depicted in figure 3.13, and participants will select the direction they believe the robot will travel towards.

3.4.4.2 Robotic Education for Kids

we are exploring how an AR robotic platform performs as a teaching tool for robotic education for kids. We applied our SENSAR framework to work with an EV3 robot as seen in figure 3.17. Here, the EV3 robot contains additional sensors that can be projected to give kids a visual aid on key concepts to robots while interacting with their robot. The next steps are to deploy our system and observe how students in small groups interact with the novel system and collaborate with each other to accomplish learning objectives.

3.4.4.3 Robotic Debugging

Figure 3.16 shows a sequence of images during two cases when the robot was navigating through a building to reach a specific destination. In the top set of images

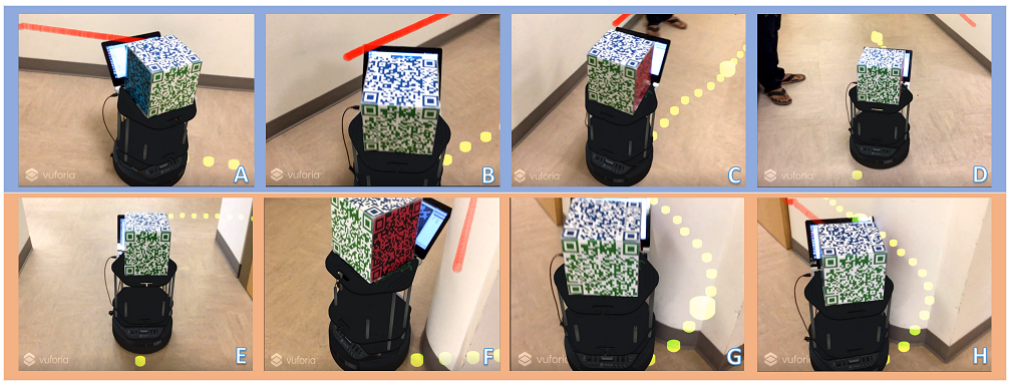


Figure 3.16: Planned path trajectory (yellow markers) with operating laser scan (Red markers) resulting in obstacle avoidance (top row). Robot operating with unexpected malfunctioning laser scan leading to wall collision (bottom row). laser scan did not detect the wall as the robot planned a right turn in image E. The lack of red markers in image E but present in the following image F provided insight that the laser scan is the source of the robot's behavior.

(A-D), the robot operated with a functioning LIDAR sensor, as seen by the red markers detecting the nearby wall. The path-planner algorithm incorporates the laser scan data with the map of the building to determine available routes to the destination while avoiding obstacles. Here, the robot successfully avoids the wall to continue down the hallway. In the bottom set of images (E-H), the robot operated with a malfunctioning LIDAR sensor, as evidenced by the absence of red markers in image E indicating that the wall is not detected and therefore is not an obstacle for the robot. As a result, the robot re-calculates a new shorter path, and the robot thus collides into the wall. SENSAR provided the user with insight into the reason for the collision, which would have been difficult to determine if the user had only visually observed the robot.

3.4.5 CONCLUSIONS

SENSAR represents an innovative AR-based robotic tool designed to facilitate the visual communication of cognitive and sensory data by robots. This project carries significant implications for the field of Human-Robot Interaction (HRI), introducing a novel and engaging mode of interaction. By transforming intricate data into intuitive visuals, SENSAR empowers individuals of varying expertise levels to ef-

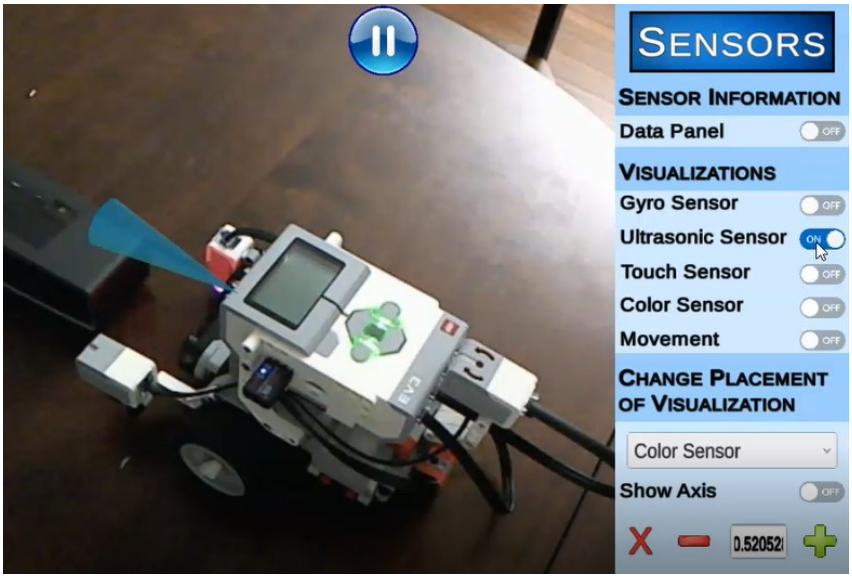


Figure 3.17: EV3 robot is projecting its ultrasonic sensor data by visualizing the blue cone that extends out towards the object detected by the robot's sensor. Other sensor can be visualized as seen on the right-hand panel.

fectively collaborate with robots. SENSAR's open-source hardware and software components have been detailed and are readily accessible through GitHub repositories. Furthermore, we have outlined potential domains where our system could prove advantageous. We anticipate that SENSAR will emerge as a valuable resource for HRI researchers, as it prompts exploration into the efficacy of visualizing diverse robot data in AR environments to enhance collaboration, learning, and reasoning capabilities.

3.5 Software: HAVEN[§§]

Due to the COVID-19 pandemic, conducting Human-Robot Interaction (HRI) studies in person is not permissible due to social distancing practices to limit the spread of the virus. Therefore, a virtual reality (VR) simulation with a virtual robot may offer an alternative to real-life HRI studies. Like a real intelligent robot, a virtual

[§§]**This chapter is based on the following paper:** HAVEN: A unity-based virtual robot environment to showcase HRI-based augmented reality. In AAAI Fall Symposium Series: Trust Explainability in Artificial Intelligence for Human-Robot Interaction (AI-HRI), 2020

robot can use the same advanced algorithms to behave autonomously. This paper introduces HAVEN (HRI-based Augmentation in a Virtual robot Environment using uNity), a VR simulation that enables users to interact with a virtual robot. The goal of this system design is to enable researchers to conduct HRI Augmented Reality studies using a virtual robot without being in a real environment. This framework also introduces two common HRI experiment designs: a hallway passing scenario and human-robot team object retrieval scenario which was later conducted as an HRI study discussed later in 3.5.7 . Both reflect HAVEN's potential as a tool for future AR-based HRI studies.

3.5.1 Introduction

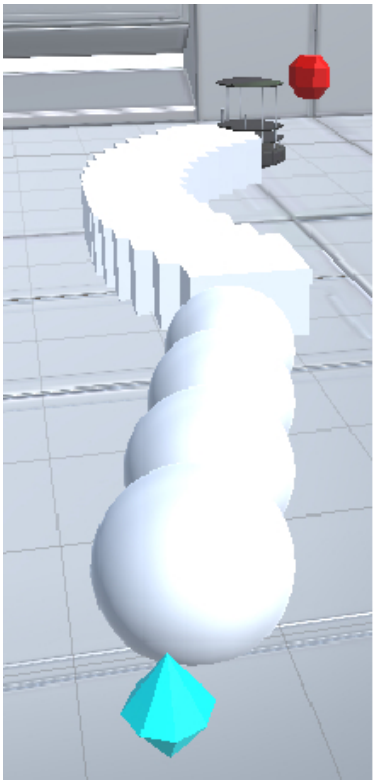


Figure 3.18: An AR visualization of the robot (Turtlebot2)'s path. White cubes represent the arc motion while the white spheres show straight motion.

To establish common ground between robot and human, the human needs to understand the robot's intention and behaviors. However, it is difficult for robots to interpret human intentions and effectively express their own intentions [184]. To

break the communication barrier, there are studies that use Augmented Reality (AR) to enable robot data that would typically be "hidden"

to be visualized by rendering graphic images over the real world using AR-supported devices (e.g., smartphones, tablets, the Microsoft HoloLens, etc.) [58]. However, AR technology requires access to a physical robot and a real-life location. When real-life HRI experiment is not accessible, a VR simulation with a virtual robot offers an alternative way to conduct HRI studies remotely.

VR is a computer-generated simulation where a person can interact within an artificial 3D environment. In VR based HRI, a virtual robot and graphic images can resemble the presence in a real-life location [53]

There are several advantages to using a virtual robot. A virtual robot can use advanced algorithms to behave autonomously like a real robot. For example, studies have compared the interactive behaviors of robots in a VR environment and a real environment. The results confirm that the behaviors of a virtual robot and a robot in a real environment are the same [112]. Virtual robots can also reduce the cost of experiments. Since multiple participants can log in to the VR environment and interact with the virtual robot at the same time, this can minimize the resources needed to conduct the studies in a real environment [112].

We introduced HAVEN (HRI-based Augmentation in a Virtual robot Environment using uNity), a VR simulation that enables users to interact with a virtual robot. We use *Unity*[†††], a game development engine, to create an interactive VR environment containing a virtual robot and its sensory data and cognitive output that is portrayed with a different type of graphic visualizations. In a technical demonstration of the system, we show that we can create a virtual robot that can act autonomously with advanced algorithms relying only on Unity. We also demonstrated two popular experimental studies in HRI, such as a hallway passing scenario and a human-robot object retrieval scenario.

[†††] https://unity.com/

3.5.2 RELATED WORK

AR provides computer-generated graphics on a user's view of the real world. In the context of mobile robots, the AR robotics system enables users and robots to communicate through a shared reality space that merges the robot's visual world with the human's [117]. The system consists of different visualizations that represent sensory and cognitive data that is rendered on the user interface. Other studies use AR interface to support human-multi-robot collaboration, where the robots collaborate with both the robot and human teammates [27].

Virtual robots have been used in VR-based HRI studies. For example, previous frameworks have combined both robot control (e.g., ROS) and virtual environment display and interaction (e.g., Unity) [36]. VR technology enables the reduction of cost in software development as well as other applications such as cloud-based robotics [112] and teleoperation tasks [36]. Other studies use Unity as the primary authoring tool for a functional virtual robot [109].

HAVEN is a continuation of these previous works, merging the AR graphic interface and a functional virtual robot in VR environment. Compared to the other systems, HAVEN allows the robot to leverage the AR virtualization to communicate with the users. Using only Unity 3D, HAVEN offers simplicity when it comes to building the virtual robot and the VR environment, allowing non-programmers to create VR simulation environment to fit their needs.

3.5.3 SYSTEM ARCHITECTURE AND DESIGN

HAVEN[‡‡‡] is a modular virtual environment designed to support many different aspects of robotic AR simulation. The functional capabilities of HAVEN include:

- Simulate various robot agents of choice. Currently the TurtleBot2 robot is used.
- Host a variety of HRI scenarios. HAVEN includes Hallway Passing and Object

[‡‡‡]The HAVEN repository is available at https://github.com/tufts-ai-robotics-group/HAVEN



Figure 3.19: Examples of AR visualizations projected by the Turtlebot2 robot. Thought bubble is rendered on top. Battery indicator is in green in the middle, and the yellow arrow are used for motion intent.

Collection in single room scenarios. Different environments are in development.

- Customizable scripts for data visualization and robotic behaviors.

3.5.3.1 Software

Built fully in the *Unity 2018.4* game development engine, all robot AI and other functionality are fully implemented using *Unity's* native C# script. Researchers who have *Unity* can clone the repository and edit the source code to implement their own algorithms and virtual hardware. The final environment can either be hosted on a cloud service or participants can download a zip that itself contains the executable and all dependencies based on operating system. This flexibility in distribution allows a wide variety of potential participants across several different platforms.

3.5.3.2 Robot Agent

The desired robot will be able to be chosen from a list. The given design for these scenarios is a Turtlebot2, currently only equipped with autonomous navigation. The AI is scripted to replicate its motion closely to its real world counterpart.

Movement Scripting - The controller script of the TurtleBot2 agent is built on *Unity's* Navigation Mesh (NavMesh) System. NavMesh scans the entire environment and lays a grid atop all navigatable surfaces. NavMeshAgents can then take that grid and mark a destination so long as there exists a valid path and will start to approach it. However, the agent will simultaneously move directly toward the destination and rotate such that to face the destination. To create a more realistic movement, the TurtleBot2 first calculates the signed angle between the direction it is facing and the direction it must face to be looking at the next destination. If the angle's absolute value is above a certain threshold or the destination is too close, then the TurtleBot2 will stop, rotate until it is facing its next destination, then continue forward. Otherwise, the TurtleBot2 will diminish the angle between itself and the destination by traveling in an arc motion until it faces the destination.

Battery Level - An optional battery feature, designed to add an extra layer of depth to navigation, is included that is currently designed to decrease in charge relative to translational motion, however different actions . For mobile robots they will always keeps track of how far it is away from its charging base. There are two different modes:

- After reaching a destination, before calculating the path to the next destination the TurtleBot2 will calculate the following:
  - How far it is from the next destination
  - How far the next destination is from the base

  If the sum of those calculations exceeds the distance remaining on the battery, the TurtleBot2's next destination becomes the charging base.

From there, it will resume reaching the rest of the destinations.

- Every frame the TurtleBot2 compares the distance from the charging base and the remaining distance the battery can travel. As soon as the remaining battery life can travel the distance to the charging base, the TurtleBot2 abandons its current path and reroutes to the charging base.

3.5.3.3 Visualization Options

HAVEN is currently equipped with 4 different types of visualization: Path Projection, Turn Signals, Thought Bubbles, and Battery Level Indicator

Path Projection (Path, Steps to Project) - Given the TurtleBot2's destination and its rotation requirements, the path is broken up into small segments (the length of the designated markers) and a marker is placed on each segment. The location of each of those markers acts as an intermittent destination when given to NavMesh. The number of markers that appear on-screen at a given time can be specified as a parameter, and there are different markers for a linear and curved motion (See figure 3.18).

Turn Signals (Angle) - The sign of the angle calculated between the TurtleBot2 and the destination determines the direction the TurtleBot2 will turn or rotate, and the appropriate direction arrow will appear (See figure 3.19).

Thought Bubbles (Message string to project) - When the human agent gets close enough to the TurtleBot2, the message for the TurtleBot2 to display appears above it in a comic style thought bubble that always faces the human agent (See figure 3.19).

Battery Level Indicator (Battery Level) - Similar to the thought bubble, battery level is indicated by a 2D image on on the TurtleBot2 that always faces the human agent (See figure 3.19).

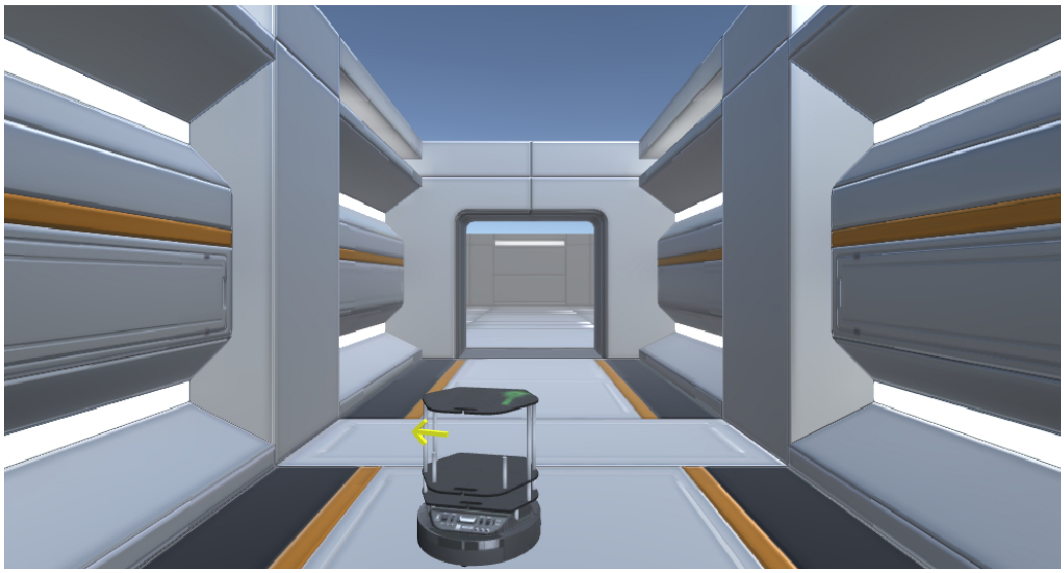


Figure 3.20: First person view of the hallway passing scenario. The TurtleBot2 detects the human (obstacle) blocking its path and therefore re-plans its path trajectory to the goal. The direction it decides to move around the obstacle is given by the rendered yellow arrow to communicate its motion intent.

3.5.4 APPLICATIONS

HAVEN simulates real-world objectives related to HRI in a format that allows for additions, modifications, and widespread distribution for various questions and research purposes. Scenarios are provided as separate *Unity* scenes built off an initial scenario template. Two example scenarios are included: Hallway traversal and Shared room object retrieval.

3.5.4.1 Hallway Passing Scenario

The Hallway Passing Scenario requires both agents to coordinate their movements in order to avoid collision with each other while attempting to reach the other side. HRI researchers have approached human prediction of robot motion intent through external robot modifications, movement behavior scripts, and many other methods. [126, 178, 54]. In HAVEN, the TurtleBot2's motion behavior can be modified using scripts, and it can use visualizations to inform the others of its decisions and intent.

TurtleBot2's goal In this scenario, the only visualization used is the turn signal.

The TurtleBot2's main objective is to reach the center of every room. The TurtleBot2 travels down the center of every hallway unless the human agent is blocking the center path, in which case the TurtleBot2 will start to curve to avoid the human, using the turn signals to indicate its desired direction (see figure 3.20).

Human's goal The human's main goal is also to reach the center of every room. In harder scenarios, the human must reach the center of a room before the TurtleBot2 does in order to remove an obstruction that sits in the TurtleBot2's path. The human will always start in a different room as the TurtleBot2 and does not know where the TurtleBot2 starts.

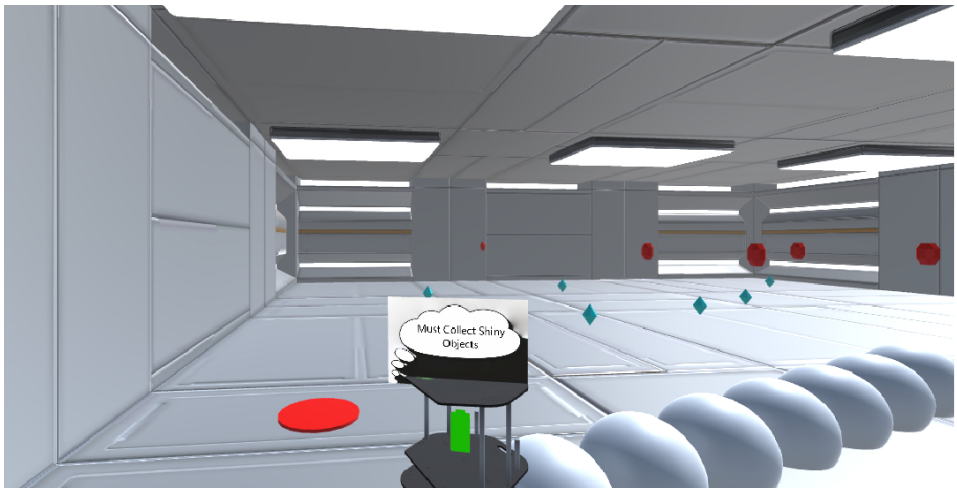


Figure 3.21: First person view of object retrieval scenario. The red circle on the floor is the TurtleBot2's charging base. The white spheres show the TurtleBot2's path trajectory. Blue gems are robot collectibles. Red gems are human collectibles.

3.5.4.2 Shared Room Object Retrieval

This scenario was inspired by [173] who conducts a study in which a participant collects beads from a set of stations with a monitor drone. Both agents share the goal of collecting their respective set of gems that are scattered around the room. The TurtleBot2 selects a gem to set its destination to and proceeds to collect it. It will not collect gems it collides with on the way. The human must stand in the gem

for a given duration before it is collected. If the human collides with Turtlebot2's path projection it will reroute to curve around the human, unless the human is within a certain distance of either the Turtlebot2 or its destination, in which case it must wait until the human moves away from the path. If the human moves away before the Turtlebot2 reaches the rerouted section it will revert back to its original path (See figure 3.21).

3.5.5 DISCUSSION

Especially given the circumstances of COVID-19, there are many benefits to working in simulation over a real-world machine. Robots and visualizations that are currently in the theoretical side can be tested, and studies can be performed. Without being restricted to a lab, participants can access and perform the study from anywhere, either hosted on the web or from downloading the executable and dependencies along with instructions in single zip file. Each study can be fully encapsulated in its own program, given sufficient instructions to participants through text prompts and manuals. The visualizations and scenarios shown in this paper are only the basics of what HAVEN can be capable of. Although not as realistically accurate as most other simulations built on more robotics-oriented software, the versatility of building solely in *Unity* removes many entry barriers for participants and can be reached by a significantly larger audience as either the technical expertise or the proper equipment are required. Not only in terms of participants, but to researchers and developers as well. With *Unity* knowledge and experience, current robots, visualizations, and scenarios can either be modified or built upon to fit specific needs or new ones can be added all together. With more control over the environment, the range of collectible data goes beyond that of robot sensory input, and many environmental variables can be controlled.

3.5.6 CONCLUSION

We introduced HAVEN, a VR simulation that enables researchers to conduct AR-based HRI studies to overcome the challenges of social-distancing mandates. With

this tool, users can interact with a virtual intelligent robot and see its sensory and cognitive output that is rendered with graphic visualizations projected over a VR world. Then we described two common popular scenarios for experimental AR studies in HRI such as the hallway passing scenarios and the human-robot object retrieval scenarios.

3.5.7 HRI Study: Dynamic Path Visualizations[§§]

The Human-robot object retrieval case study continued to a human-participant study and is described below.

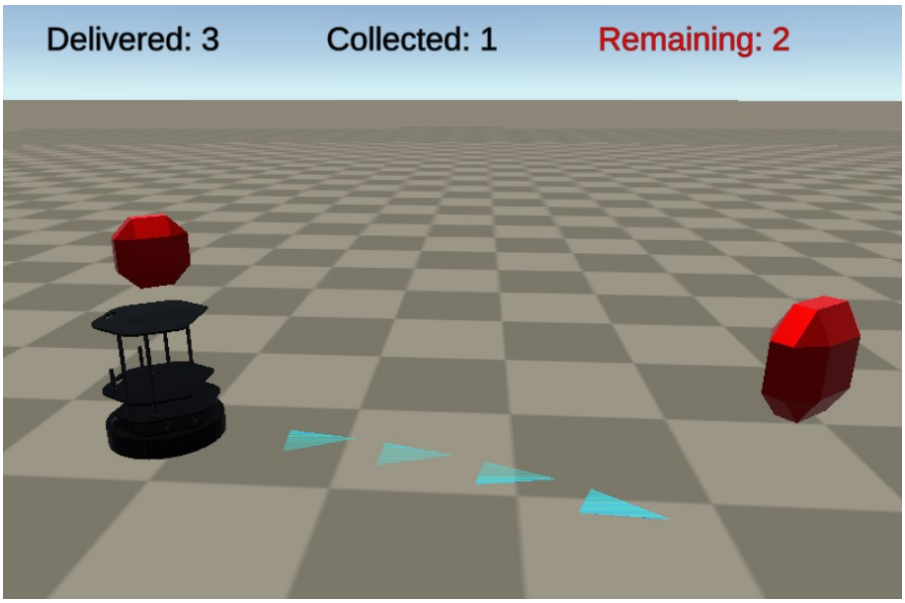


Figure 3.22: Screenshot of participant's first-person-view of a robot (Turtlebot2) projecting its path trajectory (Blue arrows) and game objects (red gems) that must be collected. Participant's progress and score are shown in text above.

Robots that cannot express intentions due to the nature of their design or environmental restrictions can be unpredictable during normal operation, resulting in undesired outcomes. These negative outcomes can be detrimental to human-robot trust when working together in a shared environment. Robots are capable of motion

[§§]**This chapter is based on the following paper:** Andre Cleaver, Darren Tang, Victoria Chen, and Jivko Sinapov. Dynamic path visualization for human-robot collaboration. In ACM/IEEE International Conference on Human-Robot Interaction (HRI) Late-Breaking Report, 2021

in a variety of ways (arm manipulators, UAVs, ground-mobile robots), and although humans can observe the robot's initial position, the final trajectory to reach their end state is mostly unknown unless time is taken to view the robot's trajectory with visual programs (e.g., Rviz). Having to switch from the physical environment to the display screen creates issues with aligning perspectives, which can be a mental burden.

To help address this problem, we can use the robot's motion plan to inform people of the robot's future behavior. For example, a ground-mobile robot has a set of destination points and the determined path trajectory to reach each point.

In this case, the robot's motion plan can be visualized in the context of the real world using AR technology. Visualizing these movements can be helpful for short actions (i.e., robot moving to the next waypoint), however, it is not clear how far into the future the robot's plan should be projected. People could find entire planned path of the robot visualized undesirable because it can be overwhelming especially if it contains several intersections or repeating laps. Our research aimed to addresses the following question: "How far into the future should a robot display its plan so that it is enough for humans to anticipate the robot's future location?"

We reasoned that if a robot's motion is simple, a person can make a reasonable guess of the robot's future location by observing its movement. However, people may find this prediction difficult if the robot's trajectory is complex. By comparing various projected path lengths, we can infer how much of the robot's future plan we need to show to improve task performance.

Our work seeks to determine if visualizing a variable amount of information regarding a robot's intended path plan (based on path complexity) improves collaborative task performance. We created a virtual game where a participant must work with an autonomous robot to complete the task of collecting gems. During the task, the robot projects its path trajectory on the ground to show the participant its intended direction of motion (See Figure 3.24). However, due to the small sample size, the quantitative evidence is limiting, but subjectively participants preferred projections of longer paths.

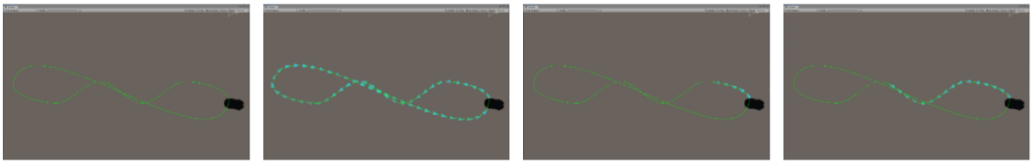


Figure 3.23: Path visual conditions from left to right: **None**, green line shows the path of the robot with no arrows, **Continuous**, blue arrows are shown along the entire path, **Fixed**, a specified amount of time is shown constantly, and **Dynamic**, a function that uses a natural logarithmic equation that takes the path's tortuosity as input to calculate the distance ahead of the robot's path to reveal.

3.5.7.1 Related Work

Communicating motion intent of a robot has been explored in several ways. Solutions include external attachments (e.g., LEDs [54, 77] and Light Projectors [150, 178]), motion patterns (e.g., flight patterns [161, 148], anticipatory motion [67, 126, 51], and deictic gestures [123, 181]), and verbal announcements [122]. Although these approaches are viable, they prevent the communication of long-term actions and potentially contribute to cognitive load. Our goal is to develop a method that conveys a robot's future navigation route in a quick and intuitive manner. We use augmented reality in simulation to evaluate our approach. By using simulation, we can render a path of any distance which may extend beyond the capabilities of a light projector.

Prior works in HRI had established that AR can be used to visualize robot data and future actions. Rosen *et al.* [144] developed a AR framework using a head-mount-display that visualizes an arm manipulator's intended motion trajectory to a goal state. Participants were asked whether the arm manipulator collides with an obstacle based on the visualized future trajectory of the arm. Quintero *et al.* [137] developed an AR-programming tool that enables users to control the actions of a robotic arm and preview the uploaded motion trajectory planner. Although both systems are successful in visualizing the arm-manipulator's future trajectory, the actions of the robot were short-term and predictable. In our study, we evaluated a robot's motion trajectory that varied in complexity using a ground-mobile robot.

Chandan *et al.* [27] developed an AR-interface that visualizes the path tra-

jectories of multiple robots. However, the participant was engaged with completing a task that was not influenced by the robots' actions; therefore, participants did not gain any advantage with the visualized paths. [173] developed an AR framework that displayed a drone path with differing levels of explicit and implicit information regarding the drone's flight planner. In their collaborative task, participants were incentivized to leverage the flight path of the drone to perform their task efficiently. Although their framework can visualize the drone's future flight path, the extent of the visuals were purposely limited to a fixed amount of time or only to a few way-point destinations. In our study, we are evaluating how the extent of a robot's future path affects users' ability to perform collaborative tasks efficiently.

Muhammad *et al.* [118] created an AR-Robotic framework that visualizes a robot's sensor and cognitive data. Two of the visual options included *Global* and *Local* Path Trajectory where the robot's entire path or a partial path is rendered respectively. Here, we determined if there is an appropriate extent of our path visualization that provides users with enough information to determine the robot's intentions but does not become overwhelming due to visual clutter [163] or cognitive load [14].

3.5.8 METHODOLOGY

We designed a 2 x 4 within-by-between human-participant experiment where participants engaged in a game with an autonomous robot. Using a component from our HAVEN simulator for online Augmented Reality HRI studies [34] [35], participants were tasked with collecting gems within a set time limit while the robot moved randomly within the same environment as the participant. Each participant could hold up to three gems at a time, and they could only deliver the gems to the robot through contact, such as moving their player towards the robot. The round ended either when all gems were collected or when 2 minutes had elapsed. In this task, we reasoned that it would be advantageous for the participant to know in advance where the robot would be to deliver the gems they had collected. We anticipated that users would leverage the path visualizations, particularly when the robot moved

in a random and unpredictable manner. Given the constant movement of the robot, participants might opt to travel in directions positioning themselves in the path of the robot. Participants without path visuals might encounter difficulty in anticipating motion intent and would miss the opportunity to position themselves ahead of the robot.

For between-subjects analysis, each participant was randomly assigned to one of four projected path visual conditions, determining the distance into the robot's future path trajectory that was visualized. These conditions were: **(1) None**, where participants saw no path; **(2) Continuous (Cont)**, where participants saw the entire path; **(3) Fixed (Fix)**, where participants saw a constant length path; and **(4) Dynamic (Dyn)**, where participants saw a changing path length dependent on the path's curvature.

For within-subjects analysis, we varied the complexity of the robot's path trajectory. The robot traveled randomly in either a **(1) simple** or **(2) complex** trajectory; therefore, each participant completed a total of 2 rounds. With the following study design, we formed the following hypotheses:

H1: Users **without visuals** will perform worse than users with visuals.

H2: Users **with visuals** will anticipate the robot's motion intent better than users without visuals.

H3: Users with **Dyn** visuals in complex path conditions will perform better than users with the other conditions.

3.5.8.1 Visualizing Path Trajectories

We can conceptualize each visual path condition as having a "render-zone", Z, measured in units of length, which reveals the robot's path trajectory to the participant. Z acts like a sliding window that starts from the robot's position and moves at the robot's speed. The length of this window depends on the selected condition. Our baseline conditions include **None** and **Cont**, that represent $Z = 0$ and $Z = \infty$ respectively. **Fix** is inspired by [173] where their *Arrow* visualization revealed 15

seconds of the robot's future path. After pilot testing with members of the lab, we selected the speed of the robot (10 units per second) and set Z to a constant value of 15 units that corresponds to 1.5 seconds. In the **Dyn** condition, Z changes according to the path's curvature or tortuosity. (i.e., a straight path yields a minimum distance projected while a curved path yields a larger distance projected). (Figure 6.1)

3.5.8.2 Z as a function of Path Complexity

We modeled the path trajectory of a robot as a curve, and the complexity of the curve is measured by a path's tortuosity. The more curvature a path contains, the harder we expect it would be for a person to predict the direction the robot will travel in the future. We calculate the path's tortuosity, τ, using the formula:

$$\tau = C/L \tag{3.1}$$

where C is the total path length, and L is the distance between the path's endpoints. Based on this equation, a straight line has $\tau = 1$, and a complete loop has $\tau = \infty$.

The robot's path was divided into an equal number of segments to calculate local tortuosities. We use this approach because it is unreasonable to represent entire path with a single curve index considering how a robot's path can change and potentially produce locations where the path curvatures are dense. We used segments of length 200 units to ensure that the segments were not so small that the tortuosity was always close to 1. Our algorithm begins with the robot's path plan, P, which is an array of waypoint coordinates:

$$P = (x_i, y_i, \theta_i)_{i=1}^{N} \tag{3.2}$$

where (x, y) is the $X - Y$ coordinate, θ is the robot's orientation, and N is the size of the array. In this study, the robot's path plan lies entirely on the X-Z plane, but

we note that it is possible to scale up a dimension to work in 3D-space. P is then used to generate a Beizer curve, and this curve is then segmented to map the local tortuosities along the robot's path. The value of Z changes as the robot makes its way along each new segment.

We used a natural logarithmic equation to calculate Z as a function of tortuosity. Our logic is that no matter how dense a path is or how close a path is to reaching a closed loop resulting in enormous values of τ that may result in Z values may reach the entire length of the path, a participant may not need the entire path visualized. A qualitative evaluation revealed that the natural logarithm equation performed well even with large values of τ. We used the following equation for our **Dyn.** condition:

$$Z = C \times ln(\tau) + B \tag{3.3}$$

where C is a scalar constant units of length and B acts a y-intercept. The lowest possible value of τ being 1 (straight path) results in a minimum $Z = B$. We chose a value of 10 and 5 for C and B respectively after a pilot test on members of the laboratory.

3.5.8.3 Experimental Implementation

Our virtual environment was created in *Unity3D* and a demo version of our game is available on *simmer.io* [§§§]. The array, P, is used as input to generate a Beizer curve that the robot will follow within the environment. We leveraged the open-source *Path-Creator* [¶¶¶] Github repository to generate the robot's path trajectories. To create the path trajectories for the robot, a grid of 25 waypoints lie within the virtual environment that is not seen by the participant. A complex path is generated by randomly ordering the sequence of waypoints that the robot will travel towards. The simple path is created using the same grid but with a reduced number of waypoints to create a relatively straighter path with fewer turns.

[§§§] https://simmer.io/@DreVinciGames/haven-test/

[¶¶¶] https://github.com/SebLague/Path-Creator

With the trajectories generated, a series of arrows are rendered along the curve at equal distances within the length of Z (see Figure 6.1: Far right). The **Dyn** condition was designed to avoid taking information away from a participant. In the case that the robot's path transitions from a curve to a straighter path, that is, Z decreases, then the extent of the visuals remain in place as the robot continues to travel along the path until the distance from the robot to the furthest visual equals the new value of Z.

3.5.8.4 Participants and Procedures

We recruited a total of 45 participants, consisting of 23 males and 22 females. The age of participants ranged from 19 to 64 (μ = 35.8,SD = 15.24). Participants reported on a 5-point scale their familiarity with robots (μ = 2.4, SD = 1.3), video games (μ = 2.8, SD = 1.6), and augmented reality (μ = 1.8, SD = 0.8). We implemented a block randomization to counterbalance the path conditions.

The entire study took place online and was accessed through a private link. The Unity application was developed on a WebGL platform and hosted on a Tufts Server. Participants controlled a virtual agent from a first-person perspective using WASD keyboard controls for movement and mouse for camera rotation.

The study required approximately 20 minutes to complete and involved the following steps:

1. Participants clicked on a pre-survey link in a web browser to determine their eligibility.

2. Participants then read the purpose of the study and instructions on how to complete it, followed by electronically signing a consent form.

3. Participants completed a game demo, which included further instructions on the task objective, navigating within the virtual environment, and collecting and delivering virtual objects.

4. Participants proceeded to the actual study, where they completed two rounds

of collecting gems.

5. Finally, participants answered a post-survey regarding their interactions with the robot.

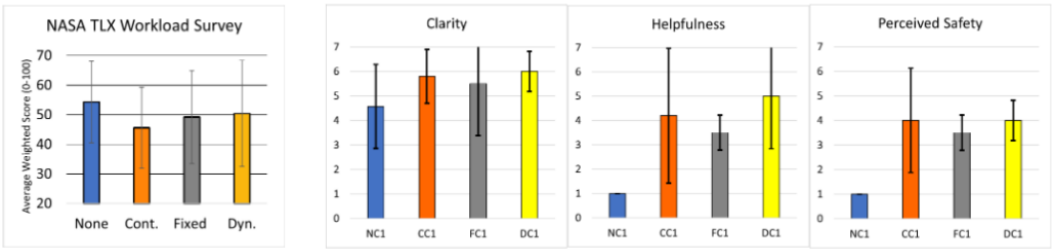


Figure 3.24: Subjective results: Averaged NASA-TLX scores (Left), Averaged ratings for Clarity, Helpfulness, and Safety(Right)

3.5.8.5 Measures & Analysis

We collected a combination of objective and subjective measurements to evaluate our four visualization options. Objective measures included the **time of completion** for each round and the **orientation** of the robot and player for each delivery. Efficient users were expected to complete their tasks in shorter times. Regarding orientation, we recorded the location and rotation of the robot relative to the player once the user made contact with the robot to deliver their collected gems. For simplicity, we focused on whether the player interacted with the anterior or posterior side of the robot. The robot was divided into two sections corresponding to its two sides (i.e., anterior and posterior) by placing contact point markers around the virtual robot. The direction of approach that the participant took to intersect with the robot triggered one of the markers. For each collision detected, a tally for the selected marker was recorded for every delivery the participant made. A tally was only counted when a participant currently held a gem; this rule was implemented to prevent any accidental contacts that could occur during the task. Markers that circled around the front of the robot (0-180 degrees) were counted towards the anterior, while the remaining side counted towards the posterior.

Subjective measures were gathered to gauge participants' perceptions and preferences using a series of 7-point Likert-style questionnaire. Scales rated "the robot was helpful in completing your task", "the robot communicated its motion intentions clearly", "how safe you expect this robot to be in a public environment?". Open-ended responses were gathered for qualitative feedback. Responses included: "Describe your experience completing the task", "Describe any strategy you used to complete the task", "Describe your experience seeing the robot's path trajectory if applicable". We then gathered the participants perceived cognitive workload using the NASA Task Load Index (NASA-TLX) questionnaire [80]. At different points throughout the study, the web interface populated a MySQL* database with collected data, which was hosted on Amazon Web Services (AWS).

3.5.8.6 Results

Figure 5.4 summarizes the average time completion for each condition per round. Unfortunately, not enough participants were recruited for this study, resulting in a low sample size per conditions. However, given the data for the complex paths for both the **None** and **Dyn** conditions, a *T-test* has not found a significant effect ($p = 0.48$). When comparing within-groups (Simple to Complex), both have shown significant effect, ($p = 0.025$), ($p = 0.027$) respectively indicating that some time is required to complete the task with more complex paths. In the simple path condition, 8 participants received the **None** condition, 2 participant for the **Cont.** condition, 3 participants for the **Fix.** condition, and 10 participants for the **Dyn.** condition. In the complex path condition, we have 2 participants in the **None** condition, 4 participants for the **Cont.** condition, 2 participant for the **Fix** condition, and 5 participants for the **Dyn** condition. The time of completion for each round is averaged as well as the participants perceived rating for Clarity, Helpfulness, and Safety.

*https://www.mysql.com/

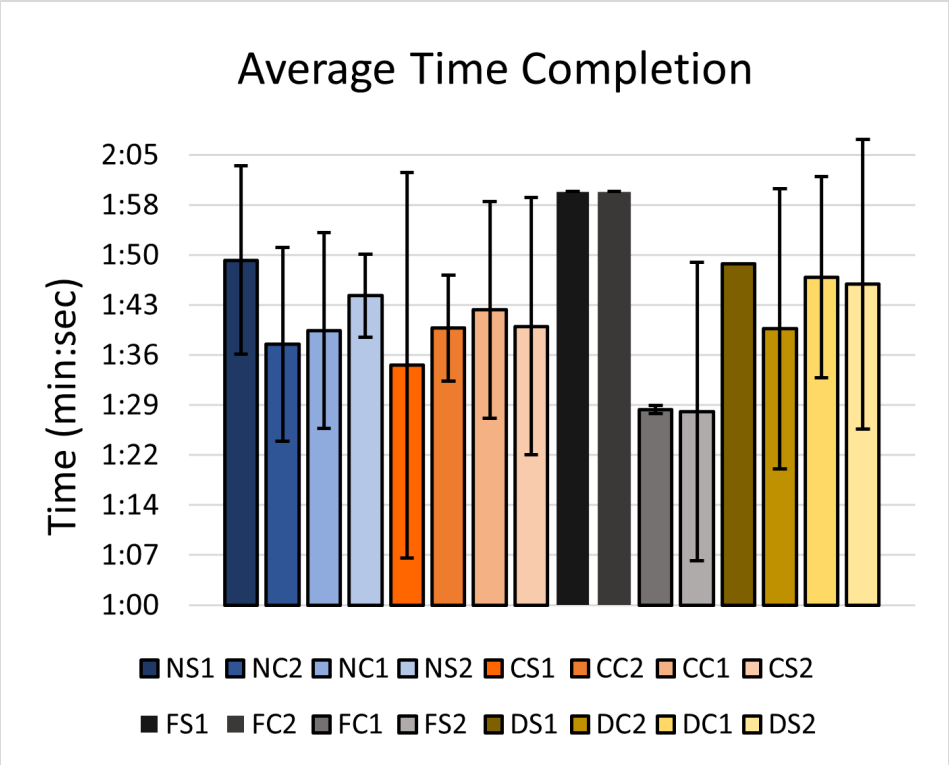


Figure 3.25: Average Time completion for each condition. First letter indicates condition, second letter indicates path condition, and the number indicates round.

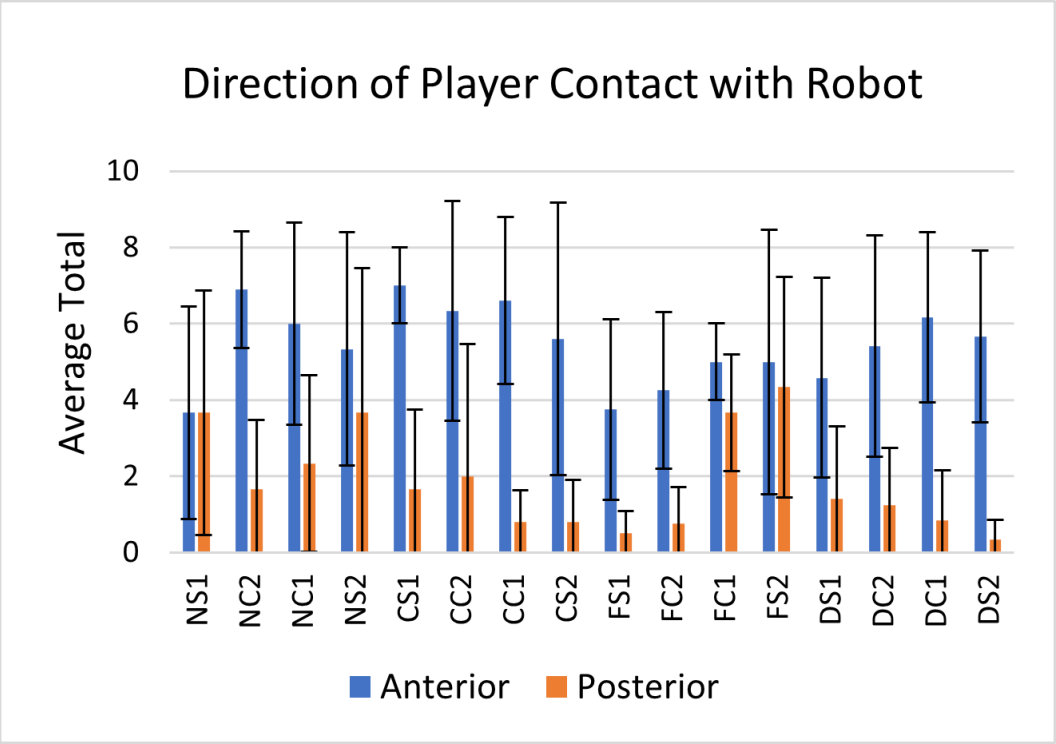


Figure 3.26: Average number of contacts made on either side of robot for each condition.

3.5.9 Discussion and Future Work

Despite our small sample size, participants' open-ended responses have revealed surprising themes regarding their strategy in completing their objective. Participants assigned **None** collected gems in random order and mostly chased the robot. Some reported anticipating the robot's path trajectory because they soon picked up on the repetitive path pattern.

> **P11:[None]** *"It was sometimes frustrating to catch the robot at first before I better figured out its movement patterns, but once I figured it out it was better..."*

Participants with visuals reported that they could wait and let the robot come to them once they positioned themselves in the projected path.

> **P12:[Cont]** *"The robot always showed its trajectory for me. So it was easy to get in front of the robot. "*

P14:[Dyn] *"The path trajectory was a great way for me to know where to go for delivery. I would just stand on the path and wait (not too long, because it's a nice short length) for the robot to run through me."*

Regarding the length of the projected robot's path, few participants expressed how they preferred having more visuals:

P19:[Fix] *"[...]it'd be nicer if you have more of the robot's path trajectory."*

Participants taking time to position themselves in the robot's path instead of continuously moving towards the robot contributed to their overall time in each round, which may have directly affected **H1**. As a result, we cannot draw conclusions for **H1**. Although no conclusions could be gathered from objective measures for now, it is worth noting that the actions of participants online are less likely to reflect the same as if this study were conducted with a physically embodied robot.

H1 reasoned that without any information on the robot's motion intent, participants will spend more time completing the task. Similarly, **H2** reasoned that participants with any information regarding the robot's motion intent will anticipate the robot's trajectory and will position themselves on the visualized path trajectory and allow the robot to collide with them on the robot's anterior side. Participants unable to anticipate the robot's trajectory cannot determine the robot's future location quick enough and could resort to chasing and contacting the robot on its posterior side.

H3 reasoned that **Dyn** will reveal a small portion of an upcoming complex path without overwhelming the user; thus, giving a participant more time to determine their next move without wasting time with a regretful action. We also wanted to consider the hypothesis that projecting the entire path (**Cont**), especially for complex paths would be overwhelming for the participant resulting in poor performance. Due to a technical error in data collection, participants assigned the **Cont** condition was not properly recorded.

We noted that participants may have had a slight advantage with the complex path, as it influences where the robot spends most of its time during the round. This means that if the robot spends a considerable amount of time in a given location due to the local density of the complex path, then the distance for the participant to travel toward the robot is shorter compared to the less complex path. It is worth exploring alternate ways to generate varying path complexities and conducting a manipulation check to determine if participants do perceive a path as predictable or not. One participant noted:

> **P21:[Dyn]** *"The first time was harder because the robot always seemed far away even when I knew its path, but the second time was easier since its path trajectory always seemed to be going towards me...*

We also want to note that several participants reported motion sickness due to the virtual robot moving around in the environment. It should be advised that any future testing with complex plans should not include fast traveling agents and repetitive paths.

For future work, we would like to investigate the effect of varying the velocity of the robot on the extent of the robot's path. Our robot's speed was held constant which allowed us to focus on the complexity of the robot's path trajectory as well as simplify the calculation of Z. With further testing, this research may provide insights to how we design future robots that are intended to operate "in the wild" and provided humans with an adequate amount of information to operate safely in a shared environment.

3.6 Summary

To summarize, the technologies used in this dissertation is a combination of robotic platforms and AR devices. By establishing a tracking system for a robot which in this case were customized multi-target images fixed to a robot, users can quickly see robotic information visualized in an intuitive manner once an AR supported device

captures those images. The aim of this research described here is explore in what contexts do people want to use AR with robots, to what extent can AR be used to help people understand the outcome of robot actions, and how should AR visuals be designed efficiently to provide people enough information regarding trajectory plans to mitigate failed outcomes. The following chapters describe the experiments conducted using the systems described here.

Chapter 4

First Encounters with a Robot: The Value of Augmented Reality for Non-Experts Performing Multiple Tasks with a Mobile Robot[§§]

4.1 Introduction

People working with a robot may find it helpful to be aware of the robot's internal representation of its environment (i.e., laser scan data) for collaborative task completion, debugging, and educational purposes. Roboticists traditionally use software like *RViz*[†] to visualize the robot's sensory and cognitive data. A drawback of RViz

[§§]**This chapter is based on the following paper:** Andre Cleaver, Victoria Chen, and Jivko Sinapov. First Encounters with a Robot: The Value of Augmented Reality when Learning about Mobile Robots. Accepted to the ACM/IEEE International Conference on Human-Robot Interaction, Workshop on Virtual, Augmented and Mixed Reality for Human-Robot Interaction (VAM-HRI 2023

[†]http://wiki.ros.org/rviz

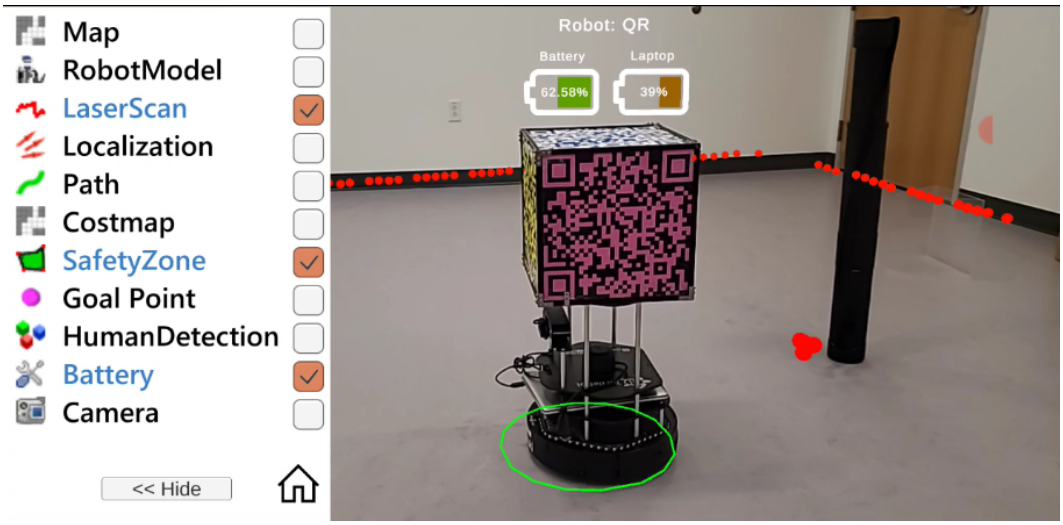


Figure 4.1: Screenshot of *SENSAR* on a tablet. The application detects the robot's pose with the target image cube and renders the selected robotic data types in relation to the cube.

is that users have to perform mental transformations on the visualized data due to the frame misalignment from the screen to the environment. To alleviate this drawback, Augmented Reality (AR) devices have been integrated with robotic systems, which can now render the data over the environment [35, 118, 34, 33, 38], ultimately minimizing any ambiguities that might arise from misaligned reference frames. However, there is a need to understand how users use AR, what value it brings, and in which types of robotic-related tasks it is most useful.

In this paper, we address this need in two parts: 1) we developed a mobile AR application, *SENSAR*, that rendered robotic data in the context of the environment (see Figure 4.1), and 2) we conducted a human-user study where participants completed robotic activities that differ in types of robotic information: *Sensors*, *Cognitive-Decisions*, *Diagnostics*, *Safety*, and *Actions*. We measured performance metrics when comparing a Control group that used only RViz with a Test group that had access to both *RViz* and *SENSAR*. We further analyzed the Test group for user preference between visualization tools.

We hypothesized that users would complete tasks with better performance using AR and that they would prefer using an AR device more in tasks related to *Sensors*, *Cognitive Decision*, *Safety*, and *Actions*. Our results show that users

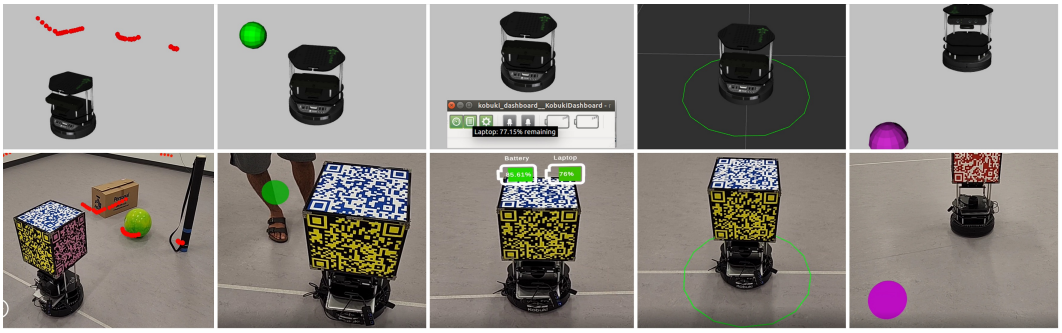


Figure 4.2: Visualizing different robotic data set types shown in RViz (top) and through an AR device (bottom). From left to right, Laserscan (Sensors), Person Detector (Cognitive), Battery Levels (Diagnostics), Safety-Zone (Safety), and Goal Point (Actions)

found value in using AR in tasks involving user movement within the environment to obtain spatially rendered data, such as the *Sensor* and *Safety* tasks. Participants who utilized the AR device in *Sensors* and *Safety* activities showed an increase in task performance compared to the Control group. Contributions in this work include our open-source AR system for Human-Robot Interaction as well as insights into the uses of AR devices in robotic training and learning.

4.2 Related Work

Research groups often develop AR robotic systems for a single type of interaction with a robot (i.e., reprogramming, collaborative task). Chong *et al.* [31], Rosen *et al.* [144], and Ong *et al.* [124] introduced an AR reprogramming system for industrial arm manipulators, allowing users to guide and plan the robot's path trajectory. Gruenefeld *et al.* [74] and Chan *et al.* [26] also used an arm manipulator to evaluate their AR safety system in collision avoidance tasks. Walker *et al.* [173] created an AR motion intent system that conveyed an aerial drone's future path trajectory in an assembly task, while Newbury *et al.* [121] visualized a robot's intent during object handovers. Williams *et al.* [183] augmented a robot's deictic gestures for guided attention tasks. Weber *et al.* [179] developed an AR calibration system to account for sensor misalignment on a robot. Xiang *et al.* [185] demonstrated a projection-based AR safety system with a mobile robot to visualize construction information

due to potential health and safety concerns with using an AR head-mounted display on a construction site. Unlike the mentioned studies, which focused on a single task, we evaluated participants in multiple types of interactions with a robot and determined if they found value in using AR devices.

We developed our *SENSAR* application to visualize multiple types of robotic data for multiple types of interaction with a mobile robot.

AR devices have also been integrated for educational robotic purposes (i.e., debugging, programming), often as part of research group studies. Pasalidou *et al.* [131] aimed to create an immersive environment to increase student engagement in robot programming activities. Radu *et al.* [139] reported the impact of unequal access to AR in group collaborative tasks with a robot in programming activities. Cheli *et al.* [30] observed students' behavior toward using the AR device while debugging their robot after unexpected behaviors. Villanueva *et al.* [171] created a robotic tele-consulting system for distance learning for students and instructors through a shared AR makerspace. Participants in these studies are often required to use the assigned AR device to complete their task. In our work, we evaluated participants' performance and preferences in completing multiple types of robotic tasks when provided the option to use either *SENSAR* or RViz.

4.3 Methodology

We designed a *2 x 5* within-and-between human-user experiment. For the between-users factor, each participant was randomly assigned to either the Control or the Test group. The Control group had access only to the computer-based RViz, the traditional robotic data visualization tool. The Test group had access to both RViz and our Augmented Reality (AR)-based version of RViz, *SENSAR*. Both visualization tools streamed the same data in terms of size, shape, color, and quantity (see Figure 4.2); therefore, the primary difference between the tools was data placement.

For the within-users factor, participants completed all 5 robotic activities, each representing different categories of robotic information: *Sensors*, *Cognitive-*

Decision, *Diagnostics*, *Safety*, and *Actions*.

4.3.1 Hardware and Software

Participants interacted with a *Turtlebot2* robot controlled using the *Robot Operating System* (ROS) [136] running on Ubuntu 16.04. The robot was equipped with an *RPLIDAR A2M8 360° Laser Scanner*. The robot's sensors and internal data were extracted and filtered from ROS using Python and C++ scripts. Participants in both the Control and Test groups had access to a desktop computer with a standard monitor screen running RViz. Participants in the Test group additionally had access to an AR device (Samsung Galaxy tablet S8) running *SENSAR*, a Unity application that rendered the robot's data in respect to the physical robot. The tablet is equipped with an LCD display with an 11-inch 2560 x 1600 resolution, with a screen ratio of 16:10. *SENSAR* followed a similar approach to visualizing data through mobile devices as described in Zea *et al.* [189].

Vuforia[*] tracked the robot using a target-image cube fixed to the robot (see Figure 4.1). Data exchange between the robot and AR device occurred over a shared Wi-fi network using *ROS-Sharp*[†]. The *SENSAR* project can be found here, *SENSAR_UNITY*[‡], *SENSAR_ROS*[§].

4.3.2 Users and Procedures

We recruited 19 participants (11 males, 7 females, and 1 who did not answer) around the University Campus. Participants' age ranged from 19 to 36 (μ=23.5, σ=4.4). Participants reported on a 7-point scale their familiarity with robots (μ=4.05, σ=2.5), visualization tools (μ=3.3, σ=2.2), and augmented reality (AR) (μ=1.85, σ=1.03). Here, the higher number means more familiarity and experience. 12 participants were selected for the Test group and 7 were selected for the Control group.

[*] https://developer.vuforia.com/

[†] https://github.com/siemens/ros-sharp

[‡] https://github.com/DreVinciCode/SENSAR_UNITY

[§] https://github.com/DreVinciCode/SENSAR_ROS



Figure 4.3: The lab space where the experiment took place with robot in the center. Objects in scene were part of the Object Detection task. The Control group used a desktop computer that remained to the side of the room (left side of image).

The study was conducted in a controlled lab setting as shown in Figure 4.3. Participants first provided consent to participate in the study after reading the study description, followed by completing a pre-questionnaire that assessed their experience with robots and AR. The lab conductor then provided a brief tutorial on how to use the visualization tools, RViz and the AR device, based on the selected condition. Participants then interacted with the robot and visual tools for a few minutes to familiarize themselves with the technology.

A reference sheet was provided for participants should they request a visual aid (see Figure 4.4). These info sheets showed the location of useful features in RViz, such as changing screen views, toggling visualized data sets, and mouse control functions. Once a participant was ready to proceed, the study continued with completing the 5 robotic activities, following the same order for each participant. Each session took approximately 60 minutes to complete the entire study.

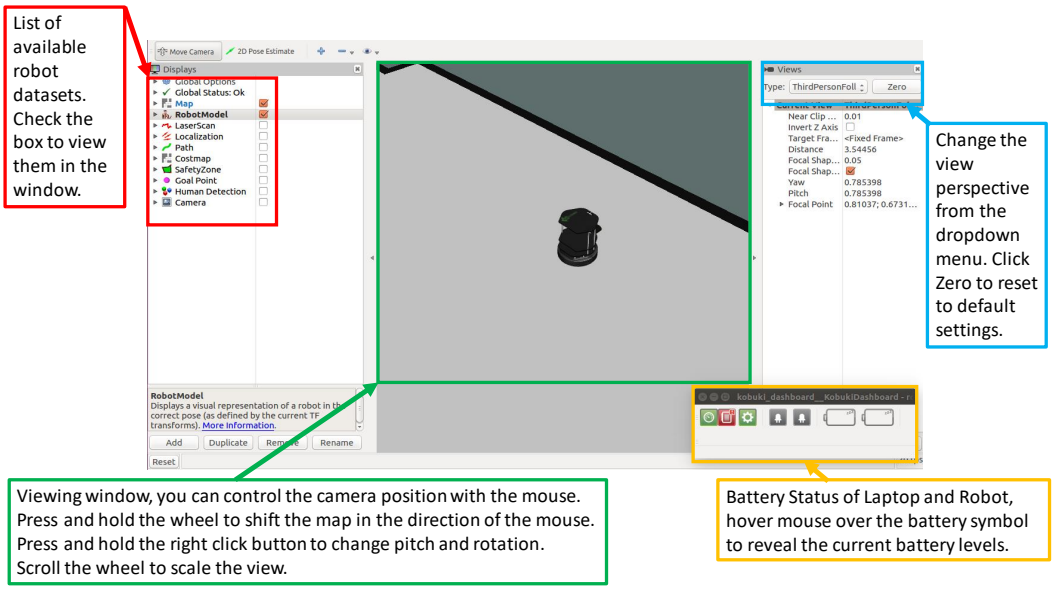


Figure 4.4: Figure shows Info Mat reference sheet provided for participants that highlights sections of RViz.

4.3.3 Robotic Activities

We list the following robotic visual categories: *Sensors*, *Cognitive-Decisions*, *Diagnostics*, *Safety*, and *Actions*, based on the taxonomy discussed by Hedayati *et al.* [82] regarding the types of information robots might convey to humans. We categorized Privacy as part of Sensors, encompassing any raw data captured by the robot. We categorized high-level information within Communication as Cognitive Decisions, which includes algorithms such as object detection and navigation planning. Condition was generalized as part of Diagnostics, encompassing the robot's internal states and specifications, such as battery life and payload capacities. Safety encompasses requirements for the robot to operate autonomously, such as an unoccupied clearance zone. The clearance zone refers to the area around the robot that must be cleared of any objects for it to continue operating. Finally, we have Actions that refer to the actions taken by the robot, such as navigating toward goals points.

As a result, we created robotic activities that we believe best represented each category. While we acknowledge that each task can be modified in terms of complexity, for this study, we intentionally kept each activity as simple as possible

so that they could be completed with either visualization tool.

The following goal oriented activities were created:

4.3.3.1 *Object Detection*

In this activity, each participant must determine which objects are detected by the robot's lidar sensor. The objects are fixed in their positions and labeled on the ground for participants to reference. Various objects of different materials, sizes, and heights are positioned arbitrarily within the room surrounding the robot (see Figure 4.3). If some items fall below the lidar sensor's plane of detection or are transparent, they will not be detected. Some objects are purposely clustered together to create ambiguity. Participants report their answers verbally to the lab conductor. We measure the total number of items correctly identified and the participants' confidence levels in completing the task. Additionally, for the Test group, we measure the total time spent on each visualization tool.

4.3.3.2 *Calibration*

In this activity, each participant was instructed to calibrate the robot's *person-detection* program. The robot determines the likelihood of a person through a leg-detection algorithm, which utilizes the robot's lidar data as input and approximates the location of that person or multiple people.

The program assigns a reliability value for each detected "person" and will only output measurements if the reliability value is greater than a threshold value. Participants were instructed to adjust the *person-detection* program by changing this threshold value.

Participants calibrated the program by adjusting a slider located on a separate laptop that corresponded to the threshold value parameter. Participants reported the threshold value that they believed best identified both the lab conductor and themselves. For this activity, we only determined if the participant was able to detect the lab conductor. Additionally, for the Test group, we measured the total time spent on each visualization tool.

4.3.3.3 *Battery Level*

In this activity, participants were required to report the robot's internal battery levels (both laptop and Turtlebot base). Robotic data in this category is not supported in visualization tools. Because there are no spatial coordinates linked with data in this category, we used the *Turtlebot_Dashboard* program that displayed the robot's battery levels in a separate computer window for the Control group. The battery levels on the AR device were rendered on top of the robot in a similar format as seen in Figure 4.2 center. For this activity, we recorded whether or not the participant was able to report the battery levels. Additionally, for the Test group, we measured the total time spent on each visualization tool.

4.3.3.4 *Safety*

In this two-part activity, participants were positioned in the middle of the room along a line, and the robot was positioned on the other end of the room, opposite the user. Participants were instructed to reposition themselves outside of the robot's "safety clearance zone" while remaining on the line as the robot moved towards the opposite end of the room. The safety clearance zone resembled a green ring located on the ground surrounding the robot (See Figure 4.1). Participants performed this task in two rounds, with the ring size changing in each round. For the second part, we gauged the participants' beliefs about the radius length of the safety ring by asking them to mark a minimum and maximum range from the robot.

Participants marked the location of the ring by placing tape on the ground while the robot was stationary. We reasoned that participants who were confident in the ring's location were likely to make shorter distances between the two markers, whereas those who were unsure of the ring's location would make larger distances between the two markers. Additionally, for the Test group, we measured the total time spent on each visualization tool.

4.3.3.5 *Goal Point*

In this activity, participants were instructed to control the robot via teleoperation and reposition the robot as close as possible to a designated goal point. The robot starts at an assigned starting position, and a single reachable goal point is marked on the robot's internal map (both in RViz and SENSAR). The Control group operated the robot using a keyboard, while the Test group had the option to use a joy stick on the tablet AR device. We recorded whether or not participants were able to successfully teleoperate the robot to the goal point. Additionally, for the Test group, we measured the total time spent using each visualization tool.
With the following study structure, we formed the following hypotheses:

> **H1**: Users with access to an AR device will complete the robotic activities more accurately than users with access to only a computer.
>
> **H2**: Users with access to an AR device will preference using the AR device more than the computer in robotic activities Sensors, Cognitive-Decisions, Safety, and Actions.
>
> **H3**: Users engaged with robotic tasks with access to an AR device will spend more time on the AR device than on the computer to complete the Sensors, Cognitive-Decision, Safety, and Action activities.

4.3.4 Measures & Analysis

To address our research questions, we employed a combination of objective and subjective measures. Video recordings with marked timestamps captured participants' interactions with the robot and visualization tools.

Objective measures for both groups included performance on the Object Detection and Safety Task. To assess performance for the Object Detection task, we calculated the average accuracy and compared it between the Control and Test groups. Accuracy was determined by dividing the total number of correctly classified objects (i.e., detected and not detected) by the total number of objects in the en-

vironment. For the Safety task, we compared the average marked distance between the Control and Test groups. The remaining tasks focused on whether participants achieved the goal. In the Test group, we measured the duration spent on each visualization tool and calculated the average percentage of total time participants used each tool.

Subjective measures included 7-point Likert-style ratings of user preference for visualization tools for each task, as well as a modified version of the Questionnaire for the Evaluation of Physical Assistive Devices (QUEAD) [146]. We also collected open-ended responses for qualitative feedback, including questions such as "Describe your experience completing the task." For the Test group, additional questions included "What prompted you to use the AR device instead of the computer?" and "What improvements would you suggest for the AR device?" We analyzed our data using T-tests to identify any significant differences between the two groups.

4.4 Results

4.4.0.1 H1

Figure 4.5 displays the average accuracy and standard deviation for the Object Detection task between the Control and Test groups. The average accuracy and standard deviation were (μ=67%, σ=13.49%) and (μ=87%, σ=7.89%), respectively. A t-test revealed significant differences between the groups (p ¡ 0.05).

The safety ring location task, in which participants marked the inside and outside of the ring, had an average length of (μ=13.28 cm, σ=6.85 cm) and (μ=4.29 cm, σ=2.42 cm) for the Control and Test groups, respectively. The large difference in averages suggests that the Control group overestimated the radius of the circle due to their uncertainty about the ring's location. A t-test revealed significant differences between the groups (p ¡ 0.05) (See Figure 4.6). All participants successfully completed the calibration, battery level, and Goal Point activities with no significant differences.

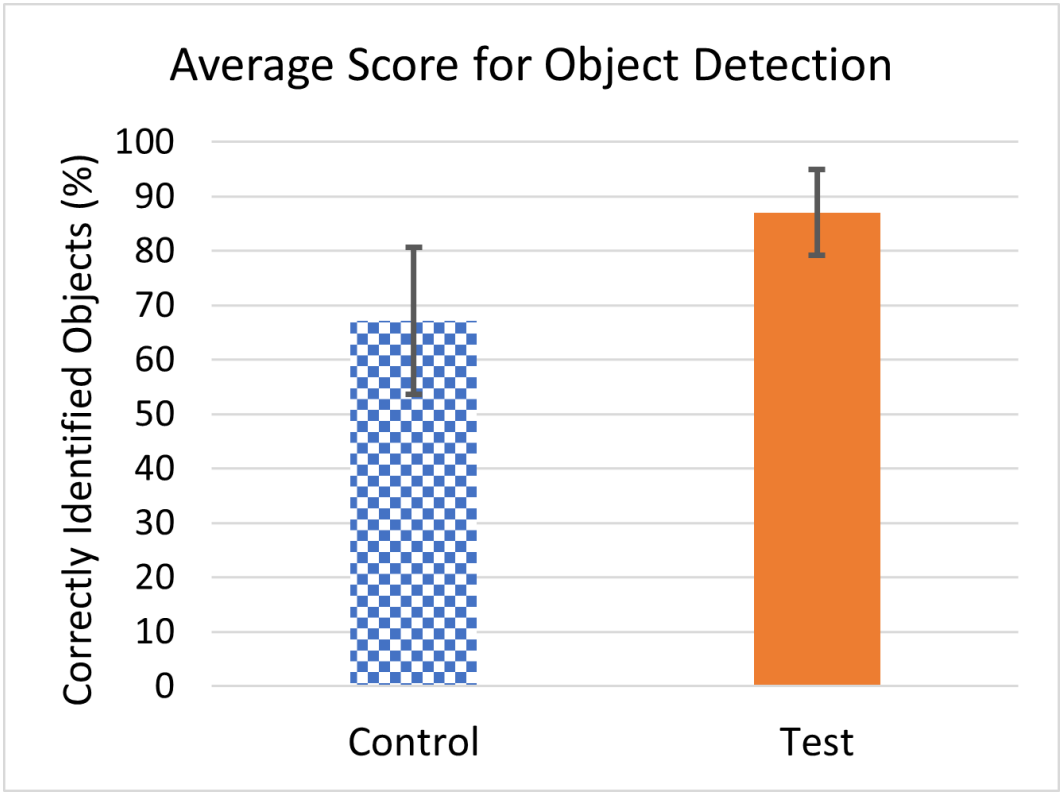


Figure 4.5: Plot of the average accuracy between the the two groups within the Object Detection Task. Solid bars indicated standard deviation.

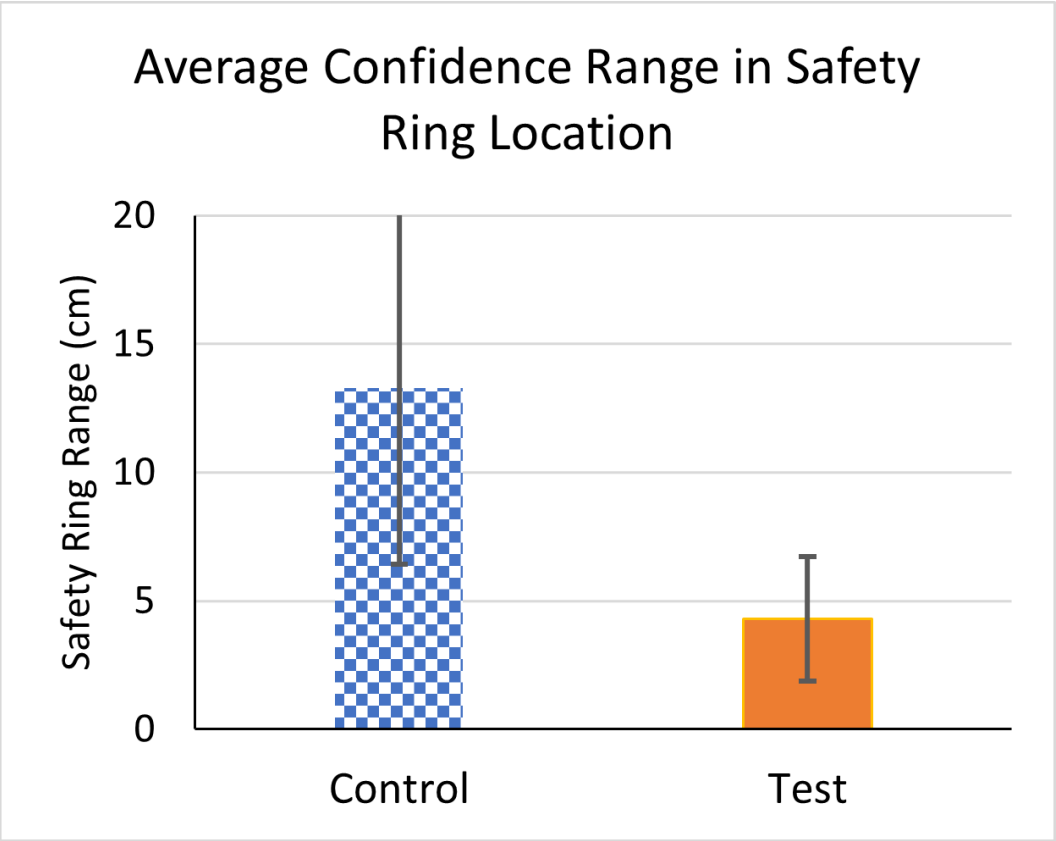


Figure 4.6: Plot of the average confidence range between the two groups within the Safety activity. Solid black bars represent the standard deviation

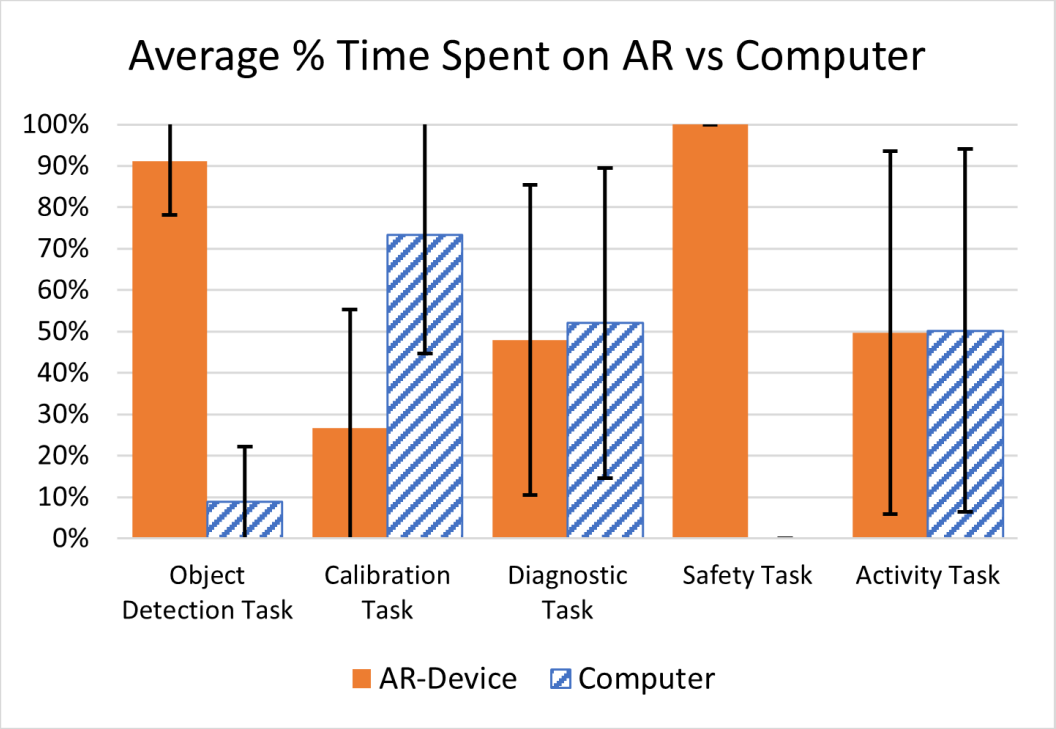


Figure 4.7: Plot of average time spent on visualization tools among the Test group for each task. Solid black bars represent the standard deviation.

4.4.0.2 H2

Figure 4.8 shows the average preference rating towards the AR device for each robotic activity within the Test group. Object Detection (μ=6.20, σ=1.13), Safety (μ=6.42, σ=1.72), and Goal Point (μ=5.67, σ=1.8) averaged towards using the AR device, while Calibration (μ=2.58 σ=1.44) and Battery Level (μ=2.75 σ=1.65) averaged more towards using the Computer. ANOVA test showed significant differences among the activities, F(4,55)= 17.5, (p<.001). Levene's test showed no significance indicating no differences in variability within the groups, (p=.457).

Figure 4.9 shows the average ratings using the QUEAD questionnaire within the Test group. Here, the higher the rating, the stronger they agree with the statement. Standard T-tests showed that there were significant differences for several items: Good Idea (t(12) = 2.53, p=.019, Likeness (t(12) = 3.09, p=.005, Rigid & Flexibility (t(12) = -4.33, p=.001, Easy to Learn (t(12) = 2.64, p=.015, Effectiveness (t(12) = 2.86, p=.009, Rapidness (t(12) = 2.47, p=.022, Efficiency (t(12) =

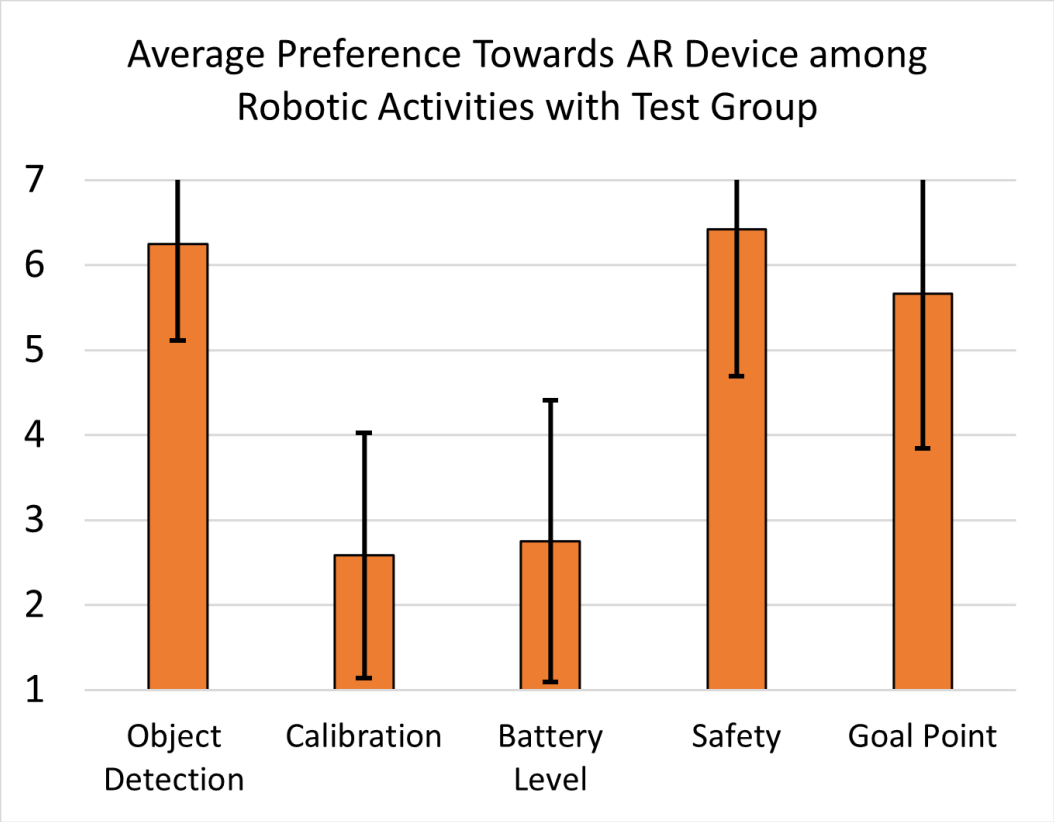


Figure 4.8: Plot of average 7 point Likert-Style on user preference towards the AR device for each robotic activity within the Test group. Solid black bars represent standard deviation.

2.30, p=.031, Performance (t(12) = 2.21, p=.038, and Usefulness (t(12) = 2.530, p=.019.

4.4.0.3 H3

Figure 4.7 shows the average time spent on each visualization tool within the Test group for each robotic activity. Here, participants mainly used the AR device for the Object Detection task. Participants mainly used the computer for the calibration activity. Diagnostic and Goal Point activities showed roughly an equal amount of visual tool usage. The Safety activity is the only task where all participants preference using the AR device. T-tests revealed significant differences for the Object Detection and calibration task (p<.001).

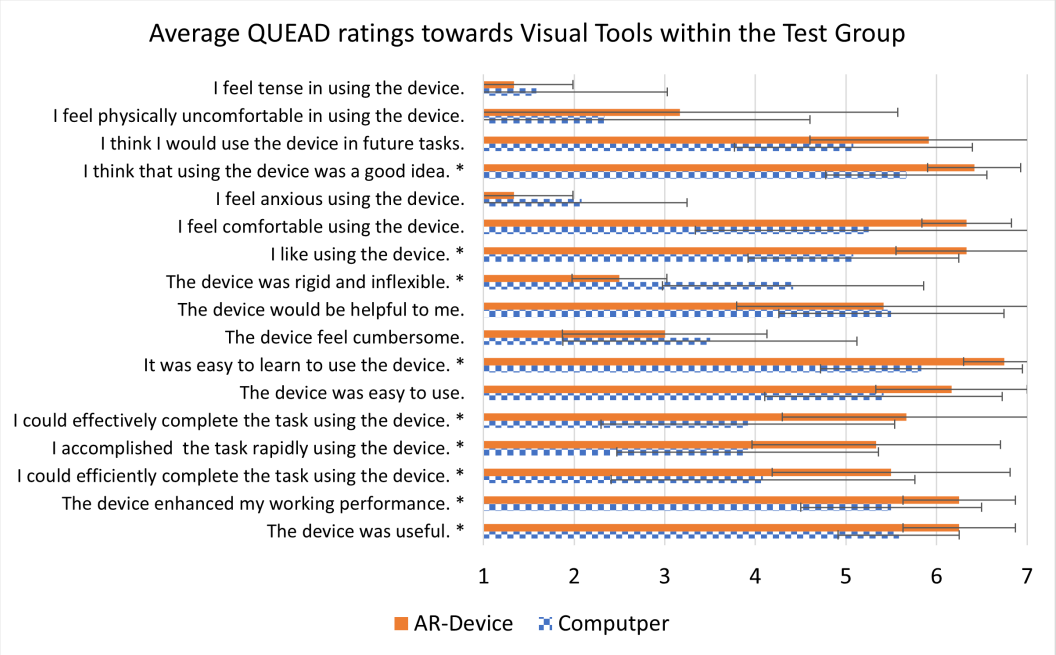


Figure 4.9: Plot of average QUAED ratings within the Test group. Asterisks indicate significant differences.

4.5 Discussion

H1 stated that users with access to an AR device will complete robotic tasks with better performance than users with access to only a computer, and our results supported this claim in the Object Detection and Safety activities. The remaining activities showed no differences. Activities Object Detection and Safety required relatively precise spatial targeting/localization of the data to perform well. AR devices reduced the ambiguity that Rviz created with the visualized data superimposed over the environment. Participant responses supported this claim:

[**Participant:7**] "*It helped you see the things you needed to see in the room instead of on a computer screen.*"

[**Participant:9**] "*It's convenient to see the where the dot actually in the real world is.*"

[**Participant:10**] "*it's easier to identify the destination using AR because you can actually stand there. You cannot achieve this using the computer.*"

In the *Action* activity however, RViz provided a top-down view of the study space, virtual model of the robot, and markers for the goal point. Nearly all participants that used RViz controlled the robot via keyboard until the virtual robot and marker aligned, all while not looking at the physical robot. Few participants reported the advantages of having a top-down view:

[**Participant:7**] "*You could basically see everything all at once instead of having to go in and out of menus*"

[**Participant:11**] "*having the overhead perspective in some cases made things easier to pinpoint, like the goal point exercise*"

[**Participant:12**] "*it's more straightforward to use the top/perspective view to capture the entire frame, and don't really need to consider obstacles in the space.*"

H2 stated that users with access to an AR device will prefer using the AR device more than the computer, and our results supported this claim for the Object Detection, Safety, and Goal Point activities. Calibration and Diagnostic information can potentially be visualized on a computer window rather than a panel in world space; therefore, users may not bother using an AR device to retrieve such information as reported by one user:

[**Participant:9**] "*For display of data not related to locations (like battery levels), it's easier if we could just see it in the computer (so we don't need to hold a tablet when checking battery levels*"

In addition, the AR device yielded better accuracy compared to the computer in almost all items in the *QUEAD* survey. When working with AR devices in the context of robotic activities, it is worth to consider the AR design interface for each task to improve user experience as one user suggested:

[**Participant:7**] "*Somehow make it so you can see all of the information without having to point the camera at the robot.*"

While our tablet could run the AR application, smooth tracking of the robot while in motion presented challenges. When the target image cube moved out of the camera's view, the visualized robot data often went out of frame, causing confusion.

To address this, we rendered virtual images only when the target image was in view. Additionally, participants in the Test group had to hold the tablet and keep the camera focused on the robot to see the visualizations. This requirement may not lead to the most comfortable or natural interaction with the robot, especially for longer durations. Some participants preferred using the computer over the tablet to avoid holding the device. Considering this, shifting towards head-mounted augmented reality devices rather than mobile AR devices may improve the user experience.

H3 stated that users will spend more time using the AR device than on the computer to complete the robotic activities, and our results supported this claim for the Object Detection and Safety activities. It is worth considering the actions of the human when deciding if an AR device is a viable visualization tool for a robotic activity. The Safety activity was the only task where participants were instructed to physically move in the environment with the robot in motion, and all participants used the AR device the entire time to accomplish that activity. During the Object Detection and Goal Point activities, we noted most users took advantage of moving within the environment with the AR device to better comprehend the visualized data, highlighting an advantage of AR.

4.6 CONCLUSIONS and FUTURE WORK

In this paper, we conducted a human-participant study to investigate the types of robotic data users visualize with augmented reality (AR) when completing tasks with a ground mobile robot. Participants were assigned tasks that varied in the type of robotic information required for successful completion, including *Sensors*, *Cognitive-Decisions*, *Diagnostics*, *Safety*, and *Actions*. In addition to traditional visualization tools, participants in our Test group had access to *SENSAR* (AR), which provided visualizations of the mobile robot's sensory and cognitive data. Our study revealed that tasks involving *Sensors* and *Safety* benefited significantly from AR devices, as evidenced by improved performance, preferences, and ratings towards AR devices. However, we observed little difference in performance and preferences

between computer-based and AR-based visualization for tasks related to *Diagnostics* and *Calibration*.

While our study highlighted the value an AR device can provide for users completing robotic activities, there are several limitations to consider. Firstly, the number of recruited participants and the size of each condition may not have been adequate for our statistical analysis. Further recruitment efforts may help address this limitation. Additionally, although we aimed to address a broad question, the results from this study have provided valuable insights for our next steps. Moving forward, we intend to delve deeper into the Sensors and Safety data categories and explore more complex tasks. We believe that these tasks will better reflect real-world scenarios encountered by factory workers who frequently interact with robots. Furthermore, it's important to note that in the object detection task, only the robot's Lidar sensor data was rendered for both conditions, and the physical objects used for detection were not represented in the virtual environment. We argue that since the robot's only output was the Lidar readings, visualizing representations of the physical objects would be considered additional information. Another limitation of our study is that our AR application relied on a mobile tablet and target images to track the robot, which required users to have the robot in view of the AR device to render any robotic information. In contrast, the computer running RViz remained stationary in one location of the room. Some users questioned whether they could move the location of the computer, which suggests a potential area for improvement in future iterations of the study.

For future work, one direction worth exploring is the visual representations used to convey robotic information. While we aimed to maintain consistency in visual representations across both visualization tools, participants may benefit from clearer and more visually appealing designs, as discussed by Groechel *et al.* [73]. Additionally, we aim to investigate how users utilize AR devices during their initial interactions with a robotic arm manipulator compared to a ground mobile robot when completing similar robotic activities. Finally, reinforcement learning (RL) in the context of robotics often remains a black box for most humans. Therefore, we

also plan to evaluate an extension of *SENSAR* [191] designed for RL and determine if it enhances human-in-the-loop RL.

Chapter 5

Helping People Predict the Outcome of Robotic Pouring Behaviors with Augmented Reality[§§]

5.1 Introduction

Individuals accumulate years of experience observing both their own and others' actions in manipulating fluids. This extensive experience equips us with an internal physics engine, allowing us to anticipate the results of a pour and intuitively identify necessary adjustments during the pouring process to prevent spills [13, 100]. However, it remains uncertain whether individuals can accurately predict the outcomes when a robot is manipulating fluids, particularly when they lack a precise mental model of the robot's capabilities upon initial encounter. Users employing an assistive robotic arm for pouring a drink may find themselves at risk of spills due

[§§]**This chapter is based on the following paper:** Andre Cleaver, Reuben M. Aronson, and Jivko Sinapov. Helping People Predict the Outcome of Robotic Pouring Behaviors with Augmented Reality. Accepted to the ACM/IEEE International Conference on Human-Robot Interaction, Workshop on Virtual, Augmented and Mixed Reality for Human-Robot Interaction (VAM-HRI 2023

to unfamiliarity with the robot's behavior.

Pouring fluids from container to container is a challenging task in robotics and is still considered an open problem requiring advanced simulators for manipulation learning [40]. Because the fluid dynamic calculations are often too difficult to solve, robots may still not apply the appropriate actions to avoid any spillage. We can leverage human ability to predict pouring outcomes and enable people to improve a robot's pouring capabilities by modifying parameters related to the manipulation process to minimize the chances of a spill. For example, a user may want to adjust the rotational speed if they foresee that the planned motion may result in a spill.

In this paper, we first want to determine to what extent visualization tools enable users to predict the outcome of a pouring action, and second, determine whether such tools enable users to adjust the planned behavior as to prevent a spill. Augmented Reality (AR) has been used to visualize a digital twin of a robotic arm manipulator simulating the pouring action corresponding to parameter changes made by the user.

The users with an assistive robotic manipulator in this case can leverage AR to provide more contextual information to help them minimize the chances of a spill before the robot executes the pour action. Another justification for utilizing an AR device lies in the fact that the conventional approach to visualizing data necessitates users to possess a display screen. This ensures the presentation of a current representation of the real-world environment, encompassing the position and orientation of the digital twin of the target container, a feature distinct from augmented reality. We hypothesized that people can better predict the outcome of the pouring action by visualizing the digital robot simulating different pouring parameters and anticipate outcomes.

We investigated how visualizing the planned trajectory of parameter changes, such as the robot's wrist performing a pouring action at different speeds, can empower people to adjust parameters of robotic pouring actions to reduce spills. We specifically ask, "Can people make reasonable predictions for a robot pouring liq-

uids better by observing a visualization of the planned action in AR and 2D displays before it is actually executed?", "Does superimposing a digital twin of the robotic manipulator over the real robot simulating pouring actions in augmented reality enable people to make more accurate guesses for the outcome of a robot pouring liquids?", and "Do users prefer AR over 2D displays when controlling the pouring parameters of a robot before executing a pour?". To answer these questions, we conducted a human-participant study where participants were instructed to assist a robot transferring fluids by manipulating the robot's pouring parameters. We hypothesized:

> **H1**: Visualizing the actions with digital twins enables users to better predict the likelihood of a spill caused by a robot manipulator pouring liquids from a source container. (control vs. AR/ 2D Displays)
>
> **H2**: Users with augmented reality to visualize robotic parameter changes will result in selecting parameters **more accurately** (**H2a**) leading to **less spillage** (**H2b**). (AR vs. control/2D Displays) The visual feedback provided by augmented reality will enable users to make more informed decisions with **less uncertainty** (**H2c**), resulting in improved control over the pouring process and minimizing the occurrence of spills.
>
> **H3**: Users will have more successful pours involving wider-rimmed containers compared to narrow-rimmed containers for all three visual conditions.

We gathered a combination of objective and subjective measures that showed spill prediction remains a challenge even with the help of visualization tools; however, AR was useful in reducing spills when setting pouring configurations for various containers especially when setting the pour angle limit and horizontal position although participants remained unsure of their decisions. Overall, participants seemed to objectively perform better with narrow containers although the difficultly in the type of container is a matter of speculation. Our AR visualization setup could further extend to other applications that require robotic systems to manipulate sensitive

substances that are deemed too dangerous for a human to handle on their own.

5.2 Related Work

Prior works related to robotic pouring have focused on developing algorithms [130, 94, 48, 95], fluid dynamics calculations [75], teaching from demonstration [145, 87, 186], and reinforcement learning [162] approaches to achieve successful pours. Although these recent works have shown promising solutions, these robotic systems require expertise to operate autonomously. Non-expert users relying on a robotic assistive device may not feel comfortable having the robot manipulating fluids unless they can supervise the process and intervene in the event the robot miscalculates a pour. In our work, we want to develop a visualization tool to enable users to make adjustments to a robots pouring behavior to enhance their confidence towards the robot making more successful pours rather than risk the robot miscalculating a pour, resulting in a spill.

Research groups have also worked to build the robot's own internal model of intuitive physics through observations of a dynamic environment [2, 176, 116] similar to how people developed this capability at an early age [78]. Studies have utilized game engines to gain insights into the core intuitive physics [170]. In our study, we explored how users can leverage their own internal visual intuitive physics engine to determine the ideal location and rotation speed to make a robot perform a pour action with the least amount of spillage.

Explainable robotics is an emerging field that aims to enable robotic systems to effectively communicate their actions, decisions, and perceptions of their environment to people which otherwise becomes a "black box" [147]. AR technology renders graphical information over the real-world with various camera devices [8, 72] and has been widely used in the Human-Robot Interaction community [160, 106, 44, 3] in applications developed for educational [30, 47], training [49], and maintenance/debugging purposes [115, 88, 89]. Specifically, research groups enabled robots to convey motion intent [45, 74, 137, 173, 144, 63], cognitive and sensory data [118,

33, 35, 34], safety information [110, 61], and affordances [135]. Relevant works have integrated robotic systems with digital twin (DT) technologies [187] to enable users within the manufacturing settings to monitor, plan, and control robots [140, 144, 28]. Work by Arboleda *et al.* focused on improving task performance of human-robot collaborative pick-and-place tasks by improving a participant's distance perception with AR [5]. Here, participants controlled a robotic arm manipulator to grasp solid objects by pressing buttons that moved the robot's end-effector and gripper orientation. Similar to their study, our participants controlled a robot to reposition the robot's end-effector to an ideal location near the target container prior to a pour action. In our study, we have participants set the robot's pour speed which cannot physically play out because such feedback will force a pour. We want to determine if participants make use of the AR visual feedback when they adjust the robot's pouring parameters to result in a minimal spill. To our knowledge, there is a lack of work exploring a human's internal physics engine with AR, as well as leveraging AR to predict robotic pouring outcomes.

One notable study by Hashimoto *et al.* [81] introduced an augmented reality device for robot control. In this study, participants interacted with a touchscreen interface that enables them to control a mobile-manipulator to complete a pick-and-place task. Using their developed augmented reality device, *TouchMe*, participants individually controlled parts of the robot to accomplish the task of placing a bottle into a trash bin. The task of transporting a bottle into a trash bin is relatively simple compared to the complexity of pouring a fluid to a target container, and their task remained the same without any changes to the environment. The participants were also evaluated only on the three scheduling methods for controlling the robot. In our study, we tasked participants to first observe the robot to determine the outcome of the robot's pour, then manipulate the robot's pouring behavior to minimize any chance of a spill.

Another notable work by Stilman *et al.*[156] developed an augmented reality tool to help resolve any ambiguities that could occur with any complex robotic system. Because a robot could contain multiple subsystems that interact within

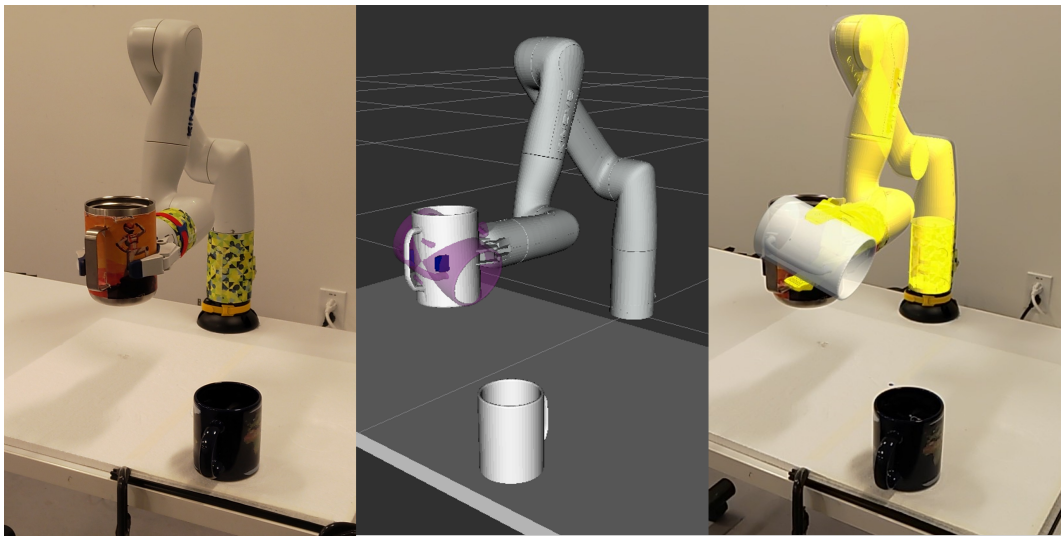


Figure 5.1: Visualization of the three conditions in our study: (Left) No Visuals/Control(CT), (Middle) RViz/2D Display(RZ) with a model of the environment, (Right) AR from the prospective of the participant. Screen capture from a Hololens2. The robot and source container digital twins are superimposed over the physical robot and container respectively.

one another, it can more difficult to determine what part of the robot's visual, navigational, or controller pipeline that is problematic in the event of an undesired outcome. By visualizing the robot's perceived ground truth, the researchers hope to identify any experimental failures and address them prior to any robot actions. The technical demonstration does not show that a user study was conducted to evaluate the effectiveness of their system. In our paper, we intend conduct a human-participant study to evaluate how visualization tools can help users identify the problem in a robot's planned actions and and essentially reprogram the robot's action.

5.3 Methodology

We conducted a 3 x 10 mixed-design study. Participants were randomly assigned to two out of three visualization conditions where each participant switched to their second assigned condition halfway through the experiment. All participants underwent 10 rounds of different common source containers divided into 2 groups where

the order of conditions were counterbalanced. Participants were instructed to help a robot pour liquid from one container to another by adjusting the robot's *pouring* parameters. Our independent variable was the method of visualizing the robot's pouring behavior.

5.3.1 Procedure

In 10 rounds, all participants completed a two-part task to assist a 6 degrees of freedom robotic manipulator with pouring water from various source containers to a target container without causing a spill. The distance between the source and target container along with the rotational angular limit and rotational velocity all determine whether or not a spill will occur. Prior to the study, the research coordinator curated a set of 10 pouring configurations that result in either a confirmed spill or a successful pour. Here, a spill occurs when water falls outside of the target container. The process of determining these configurations involved a rigorous series of trial-and-error iterations. Initially, for each container, we meticulously explored a configuration space until identifying one that consistently led to a successful pour without spillage. Subsequently, we systematically varied parameters, documenting outcomes to discern their influence on spillage occurrence. To ensure robustness, we meticulously replicated these configurations, verifying their consistency in both success and failure scenarios. Participants were notified that all containers possessed a pour configuration resulting in no spills. Moreover, certain container's pouring configurations were intentionally manipulated to induce spillage, prompting participants to adjust specific parameters for spill avoidance. Table 5.1 shows each source container configuration setup as well as a solution column to ensure no spill occurred. The Mass column is for the research coordinator to ensure the volume of water for each container is consistent for each participant.

The pouring action by the robot involved only a single joint rotation of the robot's wrist with the source container grasped by the robotic gripper. The pour action is constrained by the angular limit and rotational velocity. The angular limit is how far the robot will rotate its wrist, and the rotational velocity is how fast the

Label	Trial	Source	Mass (g)	Speed	Angle	Vert.	Horiz.	Correction
A	1	Can	200	med	120	0	-4	Angle at 120
B	7	Container	212	fast	120	0	-5	speed at slow
C	3	Blue Cup	286	fast	90	-1	-3	horizontal at -3
D	4	Plastic Bottle	264	slow	120	-2	0	speed at slow
E	5	Sports Bottle	456	slow	150	0	-4	Angle at 120
F	6	Tumbler w/ lid	280	slow	120	-2	-3	horizontal at -4
G	2	Wine	148	slow	120	0	0	horizontal at -1
H	8	Root beer	292	med	120	-1	1	vertical at 0
I	9	Paper Cup	140	slow	90	0	-3	horizontal at -2
J	10	mug	530	slow	120	-3	-5	vertical >-3

Table 5.1: Configuration table for each source container. The research coordinator references this chart to ensure the correct parameters are set prior to each participant.

robot will rotate its wrist. The position of the robotic gripper is also constrained along the x-z plane and could only move either horizontally or vertically. Translation along either axis occurs in discrete 1.5 cm lengths. For each round, the research coordinator placed a source container with water in the robot's gripper at marked heights and set the pouring parameters for the robot before the pour action. The target container remained in the same location throughout the study. The amount of water in each container was approximately half the container.

The first part of the task involved prediction of the robotic pour outcome given the source container and assigned pouring parameters, and the second part involved adjusting the robotic pouring parameters to achieve a pour without causing a spill.

5.3.1.1 Prediction

Participants first observed the scene with the robot and containers along with the current pouring parameters on a screen interface (Fig. 5.2) and visual condition assigned. We expected participants to consider several factors to make their decision on whether the robot will cause a spill during its pour action. These factors can include their intuition on the fluid dynamics of water along with the geometry of the source container, the horizontal and vertical distance between the containers,

rotational velocity, and pour angle limit. No pour action is carried out in the first part of the task. The participant's prediction was recorded and compared against the ground truth which was determined for each configuration prior to the experiment by performing the pouring action without participants in the room.

All Participants had access to a desktop monitor next to the table that displays the robot's current pouring parameters along with their assigned visualization tools. Participants complete part one of the task after reporting what portion of the ten configurations were spills or no spills along with their confidence rating.

5.3.1.2 Parameter Adjustment

In the second part of the task, participants had the opportunity to make adjustments to the robot's pouring behavior to achieve a pour without causing a spill; however, participants could only make 1 parameter change before executing the pouring action. To clarify, if the participant changed the rotational velocity that was initially *Slow* to *Fast* but later wanted to change the Pour angle limit instead, then the rotational velocity parameter returned to the original configuration of *Slow*. The pour action is then executed after the participant is satisfied with their change. The research coordinator then measured the amount of fluid transferred with a digital scale while the participant answered questions regarding their experience in their recent round.

For this study, five of the pouring configurations results in a confirmed spill unless one of the four parameters is changed to a specific value. For example, a container's rotational velocity is initially set to "fast" which purposely resulted in a spill. To avoid the spill, the participants could change the velocity setting to "slow". Changing any other parameter will still result in a spill but could vary in the amount of fluid transfer. Once the participants were content with their changes, the research coordinator executes the robot's pour action and records the outcome. Participants then answered questions regarding that container. The source container then cycles between other common containers (See 5.3.3.4 for more details).



Figure 5.2: Screenshot of the control panel used to adjust the parameters of the robot's pouring behavior .

5.3.2 Pouring Parameters

- **Rotation Velocity** - refers to the percentage of the robot's maximum rotational velocity of the robot's wrist. Participants can control the speed by selecting the speed setting (slow, medium, fast)
- **Rotation Angle Limit** - refers to the limit on how far the wrist can rotate. Users can set the angle rotation by clicking one of the 4 options (90, 120, 150, 180)
- **Horizontal Position** - refers to the robot's end effector horizontal position. By clicking either the increase or decrease buttons, the robot moves along the horizontal axis approximately 1.5 cm respectively.
- **Vertical Position** - similar to the Horizontal Position parameter but along the vertical axis.

5.3.3 Conditions

The following conditions were assigned to participants.

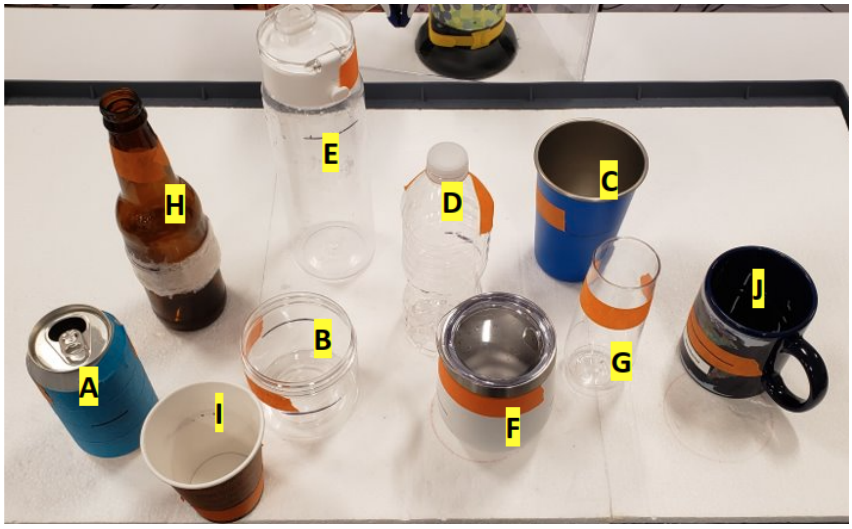


Figure 5.3: The 10 source containers used in this study.

5.3.3.1 No Visuals (Controls)

In this condition, participants had no visualization device (AR or Rviz). Participants could only rely on what they can see of the robot and the current pouring parameters. Participants could not visualize the future planned trajectory of the pouring behavior. The *No Visuals* condition evaluates **H1**.

5.3.3.2 RViz (2D Screen)

In this condition, participants were provided a standard monitor screen that runs RViz [93], a commonly used visualization tool in the robotics community. The virtual environment closely resembles the physical environment containing a digital twin of the table, robotic manipulator, and the two containers (Fig. 6.1 (E)). No fluid is simulated within the cup. Users can watch the simulated planned trajectory of the pouring behavior at any viewpoint by manipulating the camera view in RViz.

5.3.3.3 Augmented Reality (AR)

In this condition, participants used an AR device (Hololens2) to visualize the digital twin of the robotic arm and source container simulating the pouring behavior for the set pouring parameters. The visualizations in this condition mirror the robotic

arm and source container visualized in the RViz condition. As users adjust these parameters, the visualizations update to show the new expected pouring behavior. No fluid is simulated within the cup and no additional virtual objects were rendered. Simulating fluid dynamics in our AR devices is not fully supported yet and is rather difficult to ensure the same fluid simulations from RViz matches the behavior in the AR device.

5.3.3.4 Source Containers

Ten source containers were chosen as common containers that people interact with in their daily lives that was distributed equally for rim size (i.e., wide rim, narrow rim) (Fig. 5.3). For this study, the research coordinator arbitrarily selected half of the containers to require one of the four pouring parameter changes. In the final selection, container (B) required changing the speed, container (E) required an angle limit change, container (F) required changing the Horizontal Position, and containers (H and J) required changing the Vertical position, with J seen as an obvious decision due to the proximity to the source container. The containers A, C, D, G, and I did not require any changes; note that a few containers result in a pour without spill even if the vertical parameter is adjusted. All containers had markings made by the researcher to indicate the where to place each container within the robotic gripper to ensure consistency with each pour.

5.3.4 Participants

We recruited a total of 24 participants (11 males, 13 females). Participants' age ranged from 19-41 with an average of 25 (SD = 5). The study took approximately 60 minutes for each participant to complete and consisted of 5 phases: 1) Introduction, 2) Calibration, 3) Demonstration, 4) Task, and 5) Post Survey. This study was approved by the university's IRB (STUDY00004011).

In the introduction, participants, through appointment with the lab coordinator, arrived at the study space and were given a consent form which they read and signed. Participants then reported on a five-point scale [0-4] pre-questionnaire

Order	1st Round	Source Container Order					2nd Round	Source Container Order				
1	Control	A	B*	C	D	E*	RViz	F*	G	H*	I	J*
2	AR	F*	G	H*	I	J*	Control	A	B*	C	D	E*
3	RViz	J*	I	H*	G	F*	Control	E*	D	C	B*	A
4	RViz	F*	G	H*	I	J*	Control	A	B*	C	D	E*
5	Control	J*	I	H*	G	F*	RViz	E*	D	C	B*	A
6	RViz	J*	I	H*	G	F*	AR	E*	D	C	B*	A
7	AR	A	B*	C	D	E*	Control	F*	G	H*	I	J*
8	AR	J*	I	H*	G	F*	Control	E*	D	C	B*	A
9	AR	E*	D	C	B*	A	Control	J*	I	H*	G	F*
10	RViz	A	B*	C	D	E*	AR	F*	G	H*	I	J*
11	Control	E*	D	C	B*	A	RViz	J*	I	H*	G	F*
12	AR	F*	G	H*	I	J*	RViz	A	B*	C	D	E*
13	Control	E*	D	C	B*	A	AR	J*	I	H*	G	F*
14	Control	J*	I	H*	G	F*	AR	E*	D	C	B*	A
15	AR	E*	D	C	B*	A	RViz	J*	I	H*	G	F*
16	Control	A	B*	C	D	E*	AR	F*	G	H*	I	J*
17	RViz	E*	D	C	B*	A	AR	J*	I	H*	G	F*
18	AR	J*	I	H*	G	F*	RViz	E*	D	C	B*	A
19	AR	A	B*	C	D	E*	RViz	F*	G	H*	I	J*
20	RViz	A	B*	C	D	E*	Control	F*	G	H*	I	J*
21	Control	F*	G	H*	I	J*	AR	A	B*	C	D	E*
22	RViz	E*	D	C	B*	A	Control	J*	I	H*	G	F*
23	RViz	F*	G	H*	I	J*	AR	A	B*	C	D	E*
24	Control	F*	G	H*	I	J*	RViz	A	B*	C	D	E*

Table 5.2: Visual and Container order for all participants. * indicates a container that required correction to prevent spill.

their familiarity and usage with robots (M = 2.3, SD = 1.57) and augmented reality (M = 0.54, SD = 0.83). The lab coordinator then gave a brief explanation on the objective of study and the devices they were going to use. Participants who were selected to use the AR device were fitted with the Hololens2 and completed the onboard eye calibration process. In the demonstration phase, participants were shown both successful and unsuccessfully pour actions through a video and how each parameter affects the robot's pouring behavior. For those with the visualization condition, participants could see the corresponding parameter changes in RViz and/or AR device. Participants then moved on to the main task phase, described in detail in Section 6.3.5. Table 5.2 shows the order of visual condition as well as the source container condition each participant was assigned. Finally, a post survey was issued that gauged their preferences in a Likert Scale of 10 items in a seven option response format and open-ended responses.

5.3.5 Measures & Analysis

We gathered a combination of objective and subjective measures to evaluate our hypotheses. In the *Prediction* portion of the study we collected:

Task success rate in spill prediction - This measure is defined as the number of correct predictions divided by the total number of predictions.

Confidence Ratings - This measure is a seven-option response format to "How confident are you in your answer?"

In the *Parameter Adjustment* portion of the study we collected:

Percent average of fluid transfer - This measure is defined as the amount of fluid within the target container divided by the initial amount of fluid within the source container.

Parameter Selection Success Rate - This measure is defined as the number of correct parameter selections over the total number of parameter selections. It is important to note again that containers A, C, D, G, and I do not require

any parameter changes. There is a situation that we have noted where if a participant were to change the *Vertical* parameter for these containers, then the pour outcome may not result in a spill. In this scenario, we would still mark their *Parameter selection* as incorrect.

Confidence ratings - This measure is a seven-option response format to the following: "It was difficult to pick which parameter to adjust to prevent a spill.".

Open-ended Responses - This measure is an open-ended response to the following: "If a spill did occur, what would you have done differently?" and "How did you select which parameter to change?".

The post-survey included a Likert Scale with 10 Likert items with seven option response format (Cronbach's Alpha, $\alpha = 0.95$) that gauged preference between the participant's two visual conditions. Items included the following where Condition 1 and 2 are placeholders for their assigned conditions:

- (Condition 2) is easier to learn and use than (Condition 1).
- I find (Condition 2)'s interface more user-friendly compared to (Condition 1).
- (Condition 2) helped me complete my task more efficiently than (Condition 1).

Three open-ended responses aimed to gather additional information to describe their experience with the visual condition and source containers:

- "What were some signs that led you to believe that the robot was going to cause a spill or not?"
- "What other types of information about the robot or containers not provided to you in this study that could've helped you determine the outcome of the pour?"

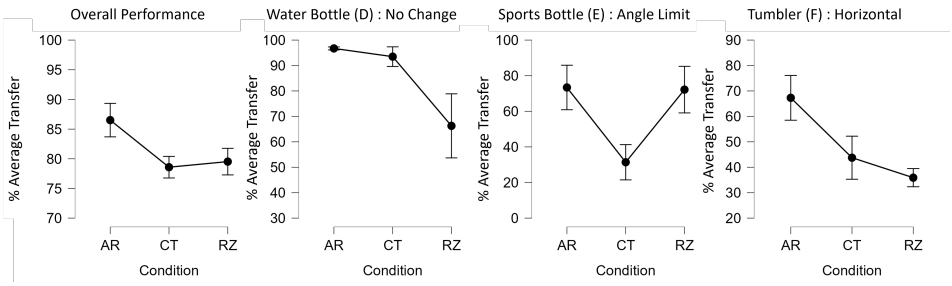


Figure 5.4: Average percentage fluid transfers by visual condition. T-bars indicate standard error.

- "Overall, what type of containers did you believe were the easiest to predict? Hardest to predict?"

We analyzed data using a one-way Analysis of Variance (ANOVA) with experimental condition of Visualization as the fixed effect. Standard Post-hoc tests, including Tukey's Correction test, compared the effectiveness across each visual condition. P-values were adjusted using the Bonferroni Correction.

5.3.6 Hardware & Software

We used a 6-DoF *Kinova Gen3 Lite* robot with a 2-finger gripper controlled with *Robot Operating System* (ROS) [136] and *MoveIt* [37] running on Ubuntu 18.04. Python and C++ scripts filtered the robotic data before delivery to the AR device and RViz [93] for visualization. Augmented visuals were developed with *Unity 2020.3.44f1*[*] and deployed onto a *Hololens2* which rendered the data over the physical environment. *Vuforia*[†] tracked the robot's position and orientation through a cylindrical target-image fixed to the robot (see Figure 7.1). A digital-twin of the robot provided by *Kinova*[‡] is placed with the target-image to match the real robot setup with C# scripts that control data exchange between the robot and AR device that occurred over a shared Wi-fi network using *ROS-TCP-Connector*[§]. Scaled digital twins of the source containers were created in modeling software and imported

[*] https://unity.com/

[†] https://developer.vuforia.com/

[‡] https://github.com/Kinovarobotics/ros_kortex

[§] https://github.com/Unity-Technologies/ROS-TCP-Connector

to both Unity and RViz.

The experiment took place in an isolated lab space that included a robotic manipulator and a target container fixed to a standard table. The amount of water filled in each source container was approximately half the container. A desktop computer connected to the robot was stationed next to the table with a tablet displaying the UI panel to control the robot pouring behavior (see Figure 5.2) along with a separate window to record participants' questionnaire responses.

5.4 Results

We analyzed the accuracy of *Spill Prediction*, i.e., the extent to which the participant was able to to correctly determine whether or not the robot will cause a spill given the set pouring parameters. *No Visuals* had an average score of 51.3% (SD = 26.3), *RViz* had an average score of 52.5% (SD = 17.7), and AR had an average score of 53.8% (SD = 22.8). We found no significant main effect $F(2, 45) = 0.05$, $p = 0.95$ rejecting **H1**.

We evaluated participant's confidence in the *Prediction* portion of the study. *No Visuals* had an average rating of 5.14 (SD = 0.64), *RViz* had an average rating of 5.09 (SD = 0.72), and AR had an average rating of 5.01 (SD = 0.67). We found no significant findings as well $F(2, 45) = 0.14$, $p = 0.87$. All source containers averaged along "Somewhat Agree" in response to the difficulty rating, "It was difficult to pick which parameter to adjust to prevent a spill."

As for the success rate of *Parameter Selection* in which the participant makes the correct parameter selection, *No Visuals* had an average score of 26.3% (SD = 15.9), *RViz* had an average score of 32.5% (SD = 16.1), and AR had an average score of 41.3% (SD = 23.6). We found a marginal main effect $F(2, 45) = 2.55$, $p = 0.09$ not fully supporting **H2a**. Confidence ratings for parameter changes showed no main effect $F(2, 45) = 0.93$, $p = 0.40$ not supporting **H2c**.

Finally, we analyzed the parameter section by container type Narrow (N) vs. Wide(W). *No Visuals* had an average score of 27.5% (SD = 29.8) and 25.0%

(SD = 29.3) , *RViz* had an average score of 32.5% (SD = 25.9) and 32.5% (SD = 19.0), and AR had an average score of 50.0 (SD = 31.9) and 32.5 (SD = 11.2) for N and W respectively. We found no significant main effect *F(2, 24) = 0.34, p = 0.71* rejecting **H3**. Figure 5.5 shows the average score on *Parameter Selection* by visual condition and container type (Narrow vs. Wide).

Figure 5.4 shows the average percent of fluid that was transferred to the target container. *No Visuals* had an average score of 78.6% (SD = 7.27), *RViz* had an average score of 79.5% (SD = 8.97), and AR had an average score of 86.5% (SD = 11.3). Overall performance revealed a significant main effect by visual condition, *F(2, 45) = 3.46, p ¡ 0.04* with an Effect size = 0.09 suggesting that the visualization condition accounts for a moderate proportion of the variance in fluid transfer. Comparing conditions of *No Visuals* and *RViz* to AR with Dunnetts's multiple comparison test, we found a decrease in performance with *No Visuals*(CT) ($p = 0.04$) supporting **H2b**.

If we look into individual containers that required specific parameter changes, we find significant effects for containers D, E, and F. Container D, which required no changes with the water bottle showed *No Visuals* had an average transfer of 93.5% (SD = 10.9), *RViz* had an average transfer of 66.3% (SD = 35.6), and AR had an average transfer of 96.7% (SD = 1.66). AR enabled users to see that the pouring parameters were adequate for a successful pour *F(2, 21) = 4.82, p = 0.02*. Post-hoc comparisons revealed that AR significantly improved perceptions of the pouring behavior over *RViz* ($p = 0.02$). Container E, which required an angle limit change with the sports bottle for a successful pour showed *No Visuals* had an average transfer of 31.4% (SD = 28.0), *RViz* had an average transfer of 72.2% (SD = 37.0), and AR had an average transfer of 73.4% (SD = 35.3) and a main effect *F(2, 21) = 4.03, p = 0.03*. Post-hoc comparisons using Tukey's Honestly Significant Difference (HSD) revealed that both AR and RViz marginally ($0.10 > p > 0.05$) improved perceptions for angular change in pouring outcomes ($p = 0.05, p = 0.06$) respectively. Container F, which required a horizontal change with the tumbler for a successful pour showed *No Visuals* had an average transfer of 43.8% (SD =

23.9), *RViz* had an average score of 35.9% (SD = 10.1), and AR had an average score of 67.3% (SD = 24.9) and a main effect, *F(2, 21) = 4.95, p = 0.02*. Post-hoc comparisons using Tukey's HSD revealed that users had a greater perception for horizontal positioning compared to RViz ($p = 0.01$) and a marginal main effect compared to *No Visuals* ($p = 0.83$).

We also ran a Sample Size calculation with a target Power = 0.8, a significance level $\alpha = 0.05$, and the effect size $\delta = 0.13$ to determine the number of participants we needed to run was approximately 68, which is well above the total number we recruited.

5.5 Discussion

There could be a number of reasons why we did not find a significant effect for **H1**. Participants on average were not highly experienced with AR or robotics as reported with the pre-questionnaire mentioned in Section 5.3.4. Because this study may be a participant's first experience with augmented reality and RViz, they may not be confident in using either device right away. The reported confidence in spill prediction averaged along "Somewhat Confident", so it is worth testing to compare

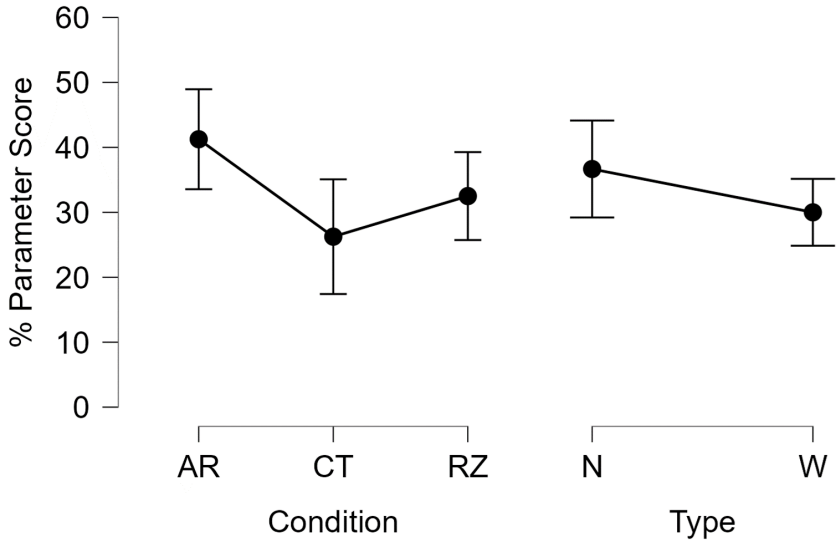


Figure 5.5: Average score in *Parameter Selection* by condition (Left) and by container type (right) where N and W correspond to Narrow and Wide respectively.

inexperienced with experienced users to see if experienced users make better spill predictions. Additionally, the pouring configurations and various containers used in our study could be too challenging for users to work with as observed in the Parameter Selection Difficulty rating. Participants have mentioned both wide and narrow containers as difficult containers to predict their outcomes. For example, one user reported the following when asked about the difficulty of the task associated with specific containers:

> [**P:6 (RZ/AR)**] *"Easier: mugs with wider openings Harder: taller bottles with lids"*
>
> [**P:7 (RZ/AR)**] "*Easiest: Tall, narrow containers (openings and water trajectory were easiest to see when animated). Hardest: Short, wide containers."*

These opposing views in participants' belief to which containers make the task easier or more difficult could explain why we did not find a significant difference for **H2c**.

Although participants were not accurate in selecting the correct pouring parameters to achieve a successful pour for (**H2a**), visuals may have helped participants position the robot to minimize the distance between the two containers ultimately allowing the bulk of the water to transfer to the target container **H2b**. We noted that most adjustments users made consisted of changing the horizontal or vertical position of the end effector.

> [**P:17 (AR/CT)**] *"For the non-ar case I based my decisions mostly off of intuition. The AR did a good job of showing how the two containers would align, using this I adjusted the robot until the two containers aligned in a way that intuition dictated would not spill."*
>
> [**P:24 (CT/RZ)**] *"The signs that I looked for were, 1) is the container positionally too close to the cup, or 2) will the pour-angle work for to pour all the liquid."*

Some participants mentioned relying on personal experience to make their decisions.

[P:12 (RZ/CT)] *"The ones [containers] I use everyday like the water bottle that you get at the supermarket and the paper and solo like cup and the mug and soda can and bottle. I don't really use a sport water bottle or a tumbler so I am not used to how they pour."*

[P:22 (CT/RZ)] *"RViz helped me mentally visualize the action and personal experience regarding the container attributes."*

A possible reason for the poor performance in the RViz condition is relative scaling of the objects within the virtual environment. Few participants suggested that the visuals in both AR and RViz(RZ) may be slightly misaligned:

[P:4 (RZ/CT)] *"No Visuals [Control] was easier for me to understand because I felt that RViz offered somewhat conflicting views between the simulation to the real robot..."*.

[P:20 (CT/RZ)] *"No visuals [Control] allowed me to rely on my intuitions when predicting a spill. In contrast, RViz seemed a bit confusing and caused me to contradict myself."*

Such misalignment could explain why differences were found for container D that did not require any changes to result in a pour without spill. However, we noticed that participants seemed to rely solely on the visuals when assigned either the AR and RViz condition. We observed a few participants of the *No Visuals* condition using their hands as a makeshift ruler to determine if the lip of the source container will reach the target container by rotating their wrists with a finger fixed at the center of the rotation point on the gripper. Although we did not expect this strategy, we allowed participants to continue and use any other strategies that they could use to make a better decision as long as they did not physically touch the robot or container.

" **[P:21 (CT/AR)]** *"The trajectory in AR was really helpful, which I could only imaging or measure by hands with No Visual."* "

This strategy of using fixed lengths to gauge distance could explain the greater performance by the No Visual condition compared to RViz. We also noticed that participants seemed to rely mostly on the visuals when assigned to the AR and RViz condition. Although they have the opportunity to view the scene, we did not record any eye tracking data to gain insights to determine where and for how long their focus was when making their decisions.

Participants highlight advantages of AR over the 2D display(RZ) and *No Visuals* conditions which contribute to the overall performance in fluid transfer:

> [**P:17 (AR/CT)**] *"AR was very useful to show where the container would end up at the end of its cycle."*
>
> [**P:7 (RZ/AR)**] *"It was useful seeing the animation directly overlaid with the object in real life, which helped me make more accurate predictions."*
>
> [**P:10 (CT/AR)**] *"Actually visualizing it was super useful, and I found myself doing that mentally for no visuals"*

Although the *Hololens2* has shown its usefulness in mental alignment as an AR device, participants have pointed out disadvantages which could have contributed to low subjective ratings.

> [**P:11 (AR/RZ)**] *"Rviz does not need calibration so it is much easier and also it is more user friendly to people wearing glasses"*
>
> [**P:10 (CT/AR)**] *"The headset felt clunky, and the depth perception made me feel a little off. No visuals felt more free"*
>
> [**P:2 (AR/CT)**] *"The headset is kinda uncomfortable"*

Participants assigned to both *RViz*(RZ) and *No Visuals*(CT) conditions stated the usefulness of having visuals demonstrating the pouring behavior.

> [**P:1 (RZ/CT)**] *"I like how I can see what the animation will look like before running it."*

[P:3 (RZ/CT)] *"In No Visuals, I could imagine the motion of the container but it was harder to visualize how that motion would occur in relation to the cup. RViz made this much easier."*

[P:22 (CT/RZ)] *"I like that RViz loops through the speed of which the robot is rotating"*

However, a few participants stated why No Visuals was more beneficial for them other than the alignment issue mentioned earlier.

[P:21 (CT/AR)] *"No setup, no calibration, easier"*

[P:5 (CT/RZ)] *"no visuals is straightforward and doesn't require looking in two places"*

5.5.1 Limitations & Future Work

Although we demonstrated a potential application of AR and 2D screens (RViz) for robotic pouring, there were some limitations. The target container was fixed relative to the robot arm along with its digital twin within RViz for simplicity. Ideally the robot should perceive both the source and target cup to account for any misalignment so that RViz displays a more accurate representation of the pouring scene. We can only say that the virtual scene created in RViz was matched to the real world as close as possible. This drawback highlights the advantage of AR by only needing to render the source container instead of the entire virtual environment. Another limitation for our study is implementing a partial within study design where participants were assigned to two out of three visual conditions. A full within participant study where each participant would see all visual conditions and source containers using balanced Latin squares to account for ordering-effects would have been infeasible due to the number of required participants; therefore, we decided to compensate with the partial within study design. We focused on controlling the visual conditions rather than the type of source containers and sacrificed order-effects among the source containers.

While participants in this study did not engage in direct interaction with the robot, we showcase how AR holds promise in enhancing users' predictive abilities regarding pouring outcomes. Future research could also delve into user engagement with the robot, directing fluid manipulation under defined pouring conditions while observing parameter alterations through AR. Another direction includes a similar study with simulating fluids as new visualizations techniques to convey the robot's pouring intent [132]. Sixty percent of participants claimed to have wanted a form of fluid simulator and water level indicator as additional useful information.

Our results showed that participants had greater performance when adjusting the angular limit and horizontal positioning; however, it is not clear if the geometry of the container played a role. It would be worth testing pouring behaviors with other container shapes such as rectangular containers [95].

5.6 Conclusions

In this work, we explored how visualizing a robot's planned actions could improve a person's prediction on the outcome of that action in the context of pouring behaviors. We used pouring as a task that is intuitive for people but challenging for robots. In a two-part task, participants first predict the outcome of a pouring behavior from various source containers and then adjust the robot's pouring parameters to reduce spillage. We found that visualizing the robot's future planned pouring trajectory in augmented reality (AR) significantly improved a participant's ability to reduce spills. Pour angle limit and horizontal positioning were two key parameters that participants were able to discern as faults and correct to avoid a spill event. However, spill prediction remained a challenge for all participants. Visualizing the planned pouring trajectory with fluid simulators in future work may improve user prediction of pouring outcomes for assistive robotics. We hope this study will inspire and inform future research about how different modes of visualization can improve the abilities of robots to interact with liquids in assistive robotics settings.

Chapter 6

Selecting Acceptable Robotic Pick-and-Place Plans with Varied Information in Augmented Reality

6.1 Introduction

Numerous social interactions, such as a restaurant server presenting a cup of water, involve implicit customer preferences regarding the execution of the task. These preferences are often unspoken but significantly influence the perceived quality of the interaction. This complexity extends to users employing assistive robots to perform similar tasks. While robots excel at task-oriented objectives, understanding and adhering to nuanced social norms can be challenging. For instance, a robot may adeptly grasp a cup and deliver it, but the subtleties of socially acceptable behavior, crucial for avoiding spills or contamination, may not be so obvious to the robot.

To mitigate undesirable outcomes, visualization tools like augmented reality (AR) offer users a preview of the robot's intended motion plans within the context of the environment. Essentially, before the robot initiates any action carrying a risk

of spillage, individuals can visually assess the planned delivery actions through AR, where a digital twin of the robot is superimposed over the physical robot. This approach aims to provide users with a comprehensive understanding of various aspects, including where the robot will grasp the cup, the directional path it will take to reach the goal state, and the placement of its delivery. Equipped with this information, individuals can assess the acceptability of the robot's plan, and if multiple plans are visualized, users can potentially choose their preferred one. User feedback becomes instrumental in guiding the robot on future cup handovers by aligning with user preferences. However, it's worth noting that users might encounter issues related to visual clutter when presented with trajectory information, as highlighted in previous studies [164].

Hence, there is merit in investigating *how* forthcoming robotic trajectory plans are presented within augmented reality (AR) to furnish users with sufficient information for informed decision-making without overwhelming them with excessive and potentially disruptive visual data. In essence, the question we aim to address is, "How can we effectively showcase robotic plans in AR to ensure that individuals can comprehend them efficiently?"

To address this question, we conducted a human-participant study where participants evaluated robotic trajectory plans designed for the successful delivery of a cup of water. All participants accessed the plans through augmented reality (AR) and were exposed to varying degrees of visual information. Within the study, we established three visual conditions, each representing a different level of trajectory information: End Points, Key Frames, and Animated. These conditions were chosen with the belief that they encapsulate *incremental* levels of information crucial for the success of a pick-and-place task involving liquid containers. Regarding the robotic plans, we selected four scenarios illustrating potential failures in a pick-and-place task to compare against our visual conditions: Grasp Region, Gripper Failure, Path Trajectory, and Speed.

The motivation behind this study stems from the balance between the time needed to process visual information and the quantity of information presented. It

could be argued that animating an entire pick-and-place plan might be the optimal way to convey information, as this visual condition offers the most comprehensive details; however, it comes at the cost of requiring users to invest the time in watching the playback of the trajectory compared to static plans, which immediately reveal key information for the user. Alternatively, focusing on visualizing static plans may cater to individuals who prioritize the immediate availability of information. With this belief, we formed the following research questions and hypotheses:

6.1.1 Research Questions

1. "What amount of robotic trajectory information is adequate enough for users to select trajectory plans that deliver items that are considered socially acceptable?"

2. "What type of visualization of robotic trajectory information do participants prefer to see to make an informed desicion?"

3. "Do users experience greater cognitive load when visualizing more trajectory information in selecting successful pick-and-place plans?"

6.1.2 Hypotheses

H1: Animated Trajectory will result in the most time spent until for the user to evaluate each plan (**H1a**) with more accurate predicted outcomes (**H1b**), while *End Points* may result in the shortest amount of time **(H1c)** but may yield in poor selected plans **(H1d)**.

H2: Users will preference seeing the entire animated trajectory condition more than the other visual conditions to make an informed decision for each plan.

H3: Users will experience the least cognitive load with *End Points* visuals while, *Key Frames* and *Animated* visual conditions will cause the most cognitive load.

In our study, we collected both objective and subjective measures that revealed participants' preferences regarding the visualization of robotic pick-and-place plans. Participants tended to favor animated representations, despite the time investment required, as they provided comprehensive information. Conversely, static renderings allowed for quicker interpretation of grasp and object placement details, facilitating the identification of potential failure points. However, employing a key frame style for visualizing robotic plans often overwhelmed participants and risked accuracy, although it did offer insights into the plan's directional path trajectory. Recognizing these nuances, a balanced approach incorporating both static and animated visual renderings emerges as potentially optimal. By doing so, we can effectively determine the outcome and acceptability of robot trajectories. Our main contribution lies in providing insights into the appropriate levels of trajectory information for robotic pick-and-place tasks involving fluids, along with the nuanced trade-offs between different visual conditions.

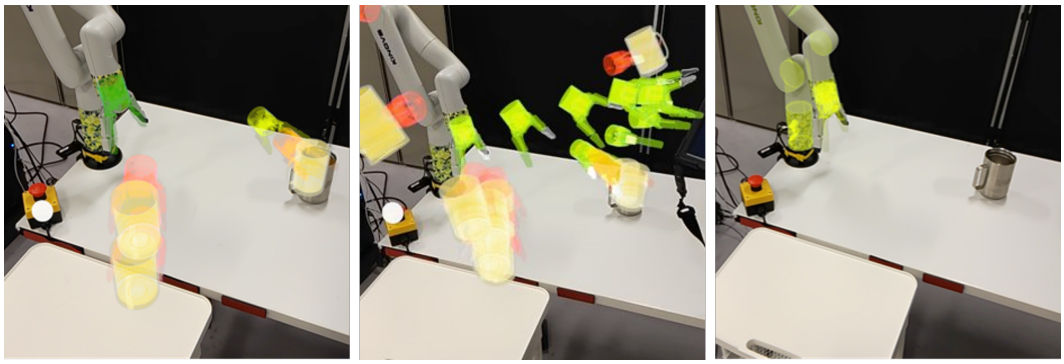


Figure 6.1: Screenshots from the point of view of a participant wearing a Hololens2 observing a robotic plan shown in the three visual conditions. The robot plan demonstrates picking up the silver cup and placing it on the adjacent tray table (Left) *End Points* condition. (Middle) *Key Frames* condition (Right) *Animated* condition.

6.2 Related Work

The exploration of human-robot interaction preferences has been a focal point in several studies, particularly concerning how robots should conduct tasks, such as delivering objects, in the presence of humans [138, 120, 125]. Sisbot *et al.* empha-

sized the significance of considering human comfort and safety in robot actions [151]. In a similar vein, Cakmak *et al.* gathered data on human preferences for robotic pick-and-place of objects, offering insights into social acceptability [21]. Building upon this foundation, our study evaluates delivery plans through Augmented Reality (AR), thereby expanding the investigation into human-robot interaction dynamics [21]. Furthermore, research has shown that humans possess the ability to predict outcomes through simulation, utilizing an internal physics engine [11]. Rosen *et al.* examined users' ability to predict robotic outcomes in AR environments, thereby laying the groundwork for our investigation into user prediction of robotic actions during pick-and-place tasks [144].

AR has emerged as a powerful tool in Human-Robot Interaction research, allowing for the overlay of graphical information onto the real world through camera devices [8, 72]. Its versatility has led to widespread adoption, particularly in educational, training, and maintenance contexts [160, 106, 44, 3]. Researchers have explored various aspects of AR interfaces, aiming to enable robots to convey motion intent, cognitive and sensory data, safety information, and affordances [45, 74, 137, 173, 144, 63, 118, 35, 34, 110, 61, 135]. Recent advancements have integrated robotic systems with digital twin (DT) technologies, facilitating the visualization of robots' future plans in AR [187]. Diehl *et al.* conducted a comparative analysis between AR and conventional 2D visualization tools, highlighting the advantages of AR in providing realistic visual feedback at a lower cost [46]. Our research extends this inquiry by investigating the efficacy of visualizing robotic plans in AR, contrasting different visual conditions to understand their impact on user comprehension [46].

However, the potential cognitive load associated with mixed-reality devices remains a concern [152, 107, 59]. Addressing visual clutter in AR displays has been a subject of investigation, with researchers proposing dynamic approaches to optimize user experience [164, 133, 33]. Furthermore, Walker *et al.* developed an AR framework for showcasing a drone's trajectory, offering insights into different modes of visualizing motion intent [173]. Our study extends this exploration by investigating alternative visualization methods for robotic arm manipulators, aiming

to enhance user prediction of outcomes.

6.3 Methodology

We implemented a within-subject study employing a 3 x 4 design. In 3 sessions, participants experienced a unique visual condition which was randomly ordered. Within each session, participants evaluated these plans all in random order that were a mix of both failed and successful plans in delivering the cup. These failed plans were grouped into 4 categories for type of failure.

6.3.1 Visual Conditions

The entire pick-and-place plan can be visualized for a user; however, not all information may be needed to make a decision on the acceptability or determine the outcome of the plan. We created three visual conditions that we believe represent varying levels of robotic trajectory information.

- *End Points (EP)* In this condition, digital twins (DT)s of the robot's gripper and target cup are rendered at static positions which are the terminal points of the trajectory plans. We can view this condition as providing the orientation and position of the robotic gripper in relation to the target cup. This condition intends to show participants how the robot is going to grasp and place the cup. With the rendering of several copies of the robots gripper, we implemented a traffic-light color gradient system to aid participants in following the flow of the entire pick-and-place plan where the start of the plan begins with the gripper rendered with the darkest shade of green and ends with the gripper with the darkest red. One important note to consider is users with color blindness and how that condition may influence their perceptions to the augmented visuals. (see Figure 6.1)

- *Key Frames (KF)* In this condition, we extend the *End Points* condition by adding more point visuals between the starting and ending goal state for each

trajectory path creating a key frame effect. This condition intends to show participants the directional path that the robotic arm will follow. The color gradient was also applied for this condition.

- *Animated (AN)* In this condition, a DT of the entire arm animates the full trajectory plan from start to goal state at a speed determined by the calculated time by the trajectory planner. We can view this condition as providing all information (speed, grasp orientation and goal state, and object grasp location). Because there are no rendered copies of the robot arm, we did not apply any color gradient to the DT. We programmed the animations to loop the entire trajectory after a 1.5 second pause between each loop.

For this study, we purposely excluded having a 2D visual tool such as RViz as one of our visual conditions. As stated in section 6.2, utilizing AR has a number of benefits over traditional visualization tools depending on the task, so for this study we are only exploring the AR space although visualizing the robotic planners on a separate 2D screen is possible.

6.3.2 Failure Conditions

Although there can be multiple points of failure for a robotic pick-and-place task, we settled on the following 4 cases which we believe are common occurrences for delivering a cup of water.

- *Grasp Failure* In this scenario, the robotic gripper fails to securely grasp the target cup (shallow reach) which results in a spill and ultimately failing the task.

- *Socially Unacceptable* In this scenario, the robotic gripper grasps the cup with one finger inside the cup ultimately contaminating the contents which may not be considered socially acceptable by the user and will fail the task.

- *Directional Path* In this scenario, the robot arm makes an arch-like maneuver with the cup securely grasped that causes the contents of the cup to spill and

thus failing the task.

- *Cup Placement* In this scenario, the robot may successfully transport the cup to the user's tray table but could ultimately fail by *how* cup is placed. A spill occurs as the robot tries to place the cup on the tray but collides with the tray before it is placed upright.

We wish to note that we initially considered the robot's *speed of delivery* as a potential factor for failure; however, we opted to exclude it to avoid adding to this study's complexity and it it likely that there are speed constraints that are imposed on robotic assistive devices for safety reasons.

6.3.3 Ground Truth Pick-And-Place Plans

In this study, robotic pick-and-place plans with the cup of water were composed of 4 sections each created by the *Moveit* planner algorithm by providing 4 goal points (waypoints). These waypoints contain both relative position and orientation that serve as the goal pose for the robot's end-effector to reach. Once provided a waypoint goal, *Moveit* outputs a plan in the form of a *Trajectory* message which essentially is instructions for each joint of the robot to move towards the goal.

The first waypoint establishes the approach trajectory, commencing from the robot's initial starting position. The second waypoint, typically located proximate to the cup, determines the gripper's grasp location on the cup. Positioned above the user's tray table, the third waypoint specifies the orientation for placing the cup. Finally, the fourth waypoint ensures sufficient clearance between the robot and the cup, preventing inadvertent collision, before returning to the home position.

The research coordinator curated a set of pick-and-place trajectories that result in either a confirmed spill or successful placement of the cup and noted the details of each execution (i.e., grasp location, trajectory curve, etc.). Each robot plan was executed for 10 trials to determine the plans success rate. Each trial determined if the robot was able to achieve 3 objectives: Does it pick up the mug, Does the mug remain in the gripper after shaking, and does the robot place the mug back on the

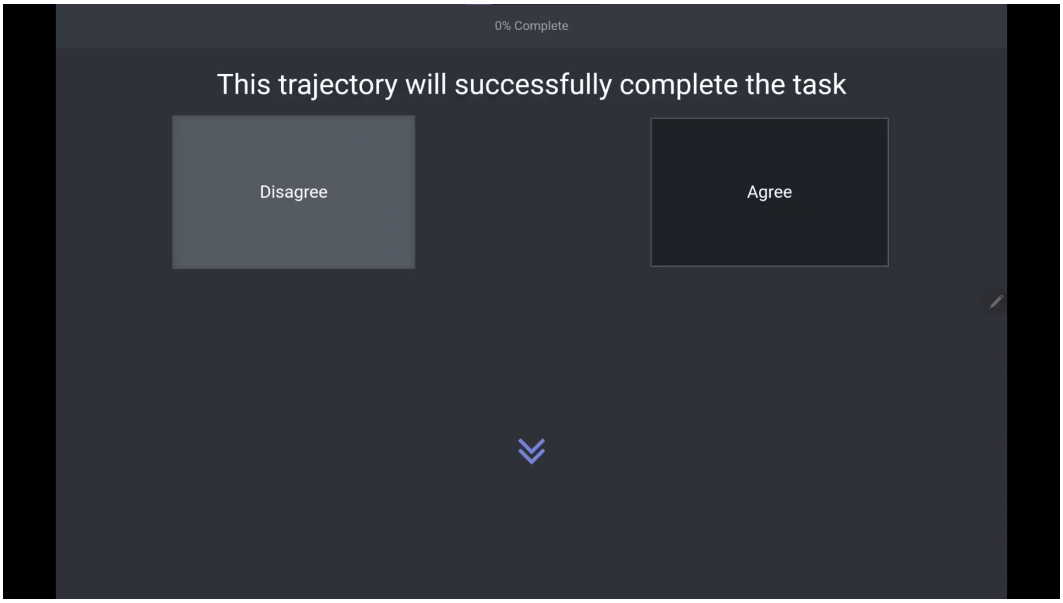


Figure 6.2: Screenshot from the tablet device where the participant answers questions regarding the current plan they are evaluating.

on table in the up-right position? We followed this process for the remaining plans, capturing a variety of scenarios. Some examples, we included plans where the robotic arm fails to grasp the cup due to the orientation of the gripper, plans that resulted in spills during the delivery, plans that include grasping the cup at different regions, and plans that place the cup at various orientations. Determining whether a plan was socially acceptable or not can depend on the participant; therefore, we noted the events that occurred with each playback and used plans that were consistent in outcome. We only considered plans that were consistent in their outcomes at least 90% of the time.

6.3.4 Participants

We recruited a total of 48 participants (21 males, 25 females, 1 Preferred not to answer). Participants' age ranged from 18-49 with an average of 26.16 (SD = 5.76). The study took approximately 45 minutes for each participant to complete and consisted of 5 phases: 1) Introduction, 2) Calibration, 3) Demonstration, 4) Task, and 5) Post Survey. This study was approved by the university's IRB (STUDY00003606).

In the introduction, participants, through appointment with the lab coordi-

nator, arrived at the study space and were given a consent form which they read and signed. Participants then reported on a five-point scale [0-4] pre-questionnaire their familiarity and usage with robots (M = 1.85, SD = 1.62) and augmented reality (M = 0.39, SD = 0.64, and video gaming experience (M = 2.25, SD = 1.32).

The lab coordinator then gave a brief explanation on the objective of study and the Augmented Reality device (Hololens2) they were going to use. Participants were fitted with the Hololens2 and completed the onboard eye calibration process. In the demonstration phase, participants were shown an example of a robotic plans that only aimed to pick and raise the cup in all three visual conditions through a prerecorded video.

Participants then moved on to the main task, described in detail in Section 6.3.5 and a post survey was issued at the conclusion of the study that included a rank-order question and 4 open-ended responses that gauged their experienced with the visual conditions. At the end, each participant was compensated with an Amazon gift card valued at $15 USD.

6.3.5 Task Description

In this study, participants were tasked with evaluating plans executed by a robot tasked with a pick-and-place goal with a cup of water onto their tray table. We explained to participants that the robot is acting as a robotic waiter and is learning how to deliver a cup of water from a table onto a tray. Participants were briefed that the robot had identified multiple approaches for this task, but the robot is uncertain regarding the success of those approaches. Participants were assigned the role of evaluating and providing "feedback" on the robot's delivery plans. Their evaluation involved assessing the plans' suitability for execution and predicting potential outcomes, particularly in scenarios where the plans might encounter failure. Throughout the study, participants experienced all visual conditions and potential failure conditions through 3 sessions. Participants responded to questions via tablet to assess their beliefs about the outcome and its acceptability for each plan. Note here, that no physical attempt of grasping the cup of water takes place during the

study. Participants were only evaluating the curated plans created and tested by the research coordinator.

During the main task of evaluating the assigned plans, participants concurrently engaged in a Detector-Response Test (DRT) as a secondary task. Adapted for our augmented reality (AR) device, the DRT methodology mirrored the approach outlined by Zaman *et al.* [188] in the realm of virtual reality, replicating the functionalities of ISO 17488:2016 originally tailored for driving assessments [157]. The DRT served two objectives: firstly, to prevent participants from memorizing robotic plan outcomes, and secondly, to offer a more objective assessment of cognitive load compared to the self-reported NASA-TLX survey [80]. In Figure 6.1, the left and middle panels depict a white dot positioned towards the left center, while the right panel excludes this dot, adhering to the protocol outlined in [157] where the dot flashes red. Each participant received a handheld mechanical button, connected to a desktop computer via USB. A Python script monitored button presses, incrementing the participant's overall score when pressed within a specified time window. After each session, we administered a NASA-TLX questionnaire to obtain a more detailed assessment of participants' cognitive levels for each visual condition. The DRT was not running during the NASA-TLX questionnaire part of the session.

6.3.6 Open-Condition Task

After completing the main task, participants transitioned to an open-condition session, mirroring the main task but affording participants the liberty to choose their preferred visual condition. Users were informed of their ability to switch conditions at any point during the session. This session featured only five new plans for participants to evaluate. New trajectories where used during this portion of the study.

6.3.7 Measures & Analysis

We gathered a combination of objective and subjective measures to evaluate our hypotheses.

Success prediction accuracy - This measure is defined as the number of correct predictions divided by the total number of predictions.

Confidence Ratings - This measure is a seven-option response format to "How confident are you in your answer?"

Elapsed Time - This measure is defined as the duration taken for a participant to determine the outcome of a plan. The time starts once the plan is projected and stops once the participant clicks a button to continue on to the next plan.

DRT Score - Response times for the DRT are measured as the time from stimuli onset until the time the user responds to it, and hit rates are calculated as the ratio of correctly answered stimuli (from 100 ms to 2500 ms) out of all presented stimuli.

NASA-TLX - A self reporting survey that assesses work load categorized into 6 subclasses on a 10 point scale [80].

Open-ended Responses - This measure is an open-ended response to the following:
What were some signs that led you to believe that the robot was going to cause a spill or not?
Describe your experience with each visual condition.
Are there any other types of information not shown in this study that you think would help you perform your task better?

At the end of the main task, we tasked the participant to complete an additional task similar to the main task where participants were able to freely select which visual conditions they wanted.

Percent Total Time for each Visual Condition - This measure is defined as the total time they spent on a visual condition over the total time spent on all visual conditions.

We analyzed data using a one-way Analysis of Variance (ANOVA) with experimental condition of Visualization as the fixed effect. Standard Post-hoc tests, including Tukey's Correction test, compared the effectiveness across each visual condition. P-values were adjusted using the Bonferroni Correction.

6.3.8 Hardware & Software

We used a 6 degrees-of-freedom *Kinova Gen3 Lite* robot with a two finger gripper controlled with *Robot Operating System* (ROS) [136] and *MoveIt* [37] running on Ubuntu 18.04. Python and C++ scripts filtered the robotic data before delivery to the AR device. Augmented visuals were developed with *Unity 2020.3.44f1*[*] and deployed onto a *Hololens2* which rendered the robotic data over the physical environment. *Vuforia*[†] tracked the robot's position and orientation through a cylindrical target-image fixed to the robot (see Figure **??**). A digital-twin of the robot provided by *Kinova*[‡] is placed with the target-image to match the real robot setup with C# scripts that control data exchange between the robot and AR device that occurred over a shared Wi-fi network using *ROS-TCP-Connector*[§].

The experiment took place in an isolated lab space that included a robotic manipulator fixed to a standard table. A desktop computer connected to the robot was stationed next to the table with a Samsung Galaxy Tab S8 mounted to opposite end of the table to record participants' questionnaire responses (Fig. **??**). A hand-held mechanical button connected to an Arduino Nano [¶] registered button presses by the user for the Detector-Response Test.

6.4 Results

We analyzed the accuracy, confidence, and elapsed-time in users' ability to predict the outcome of various failed and successful planned pick-and-place task of delivering

[*] https://unity.com/

[†] https://developer.vuforia.com/

[‡] https://github.com/Kinovarobotics/ros_kortex

[§] https://github.com/Unity-Technologies/ROS-TCP-Connector

[¶] https://www.arduino.cc/

a cup of water across *End Points* (EP), *Key Frames* (KF), and *Animated* (AN) visual representations and summarized the results in Fig. 6.3, 6.5, 6.6, 6.7.

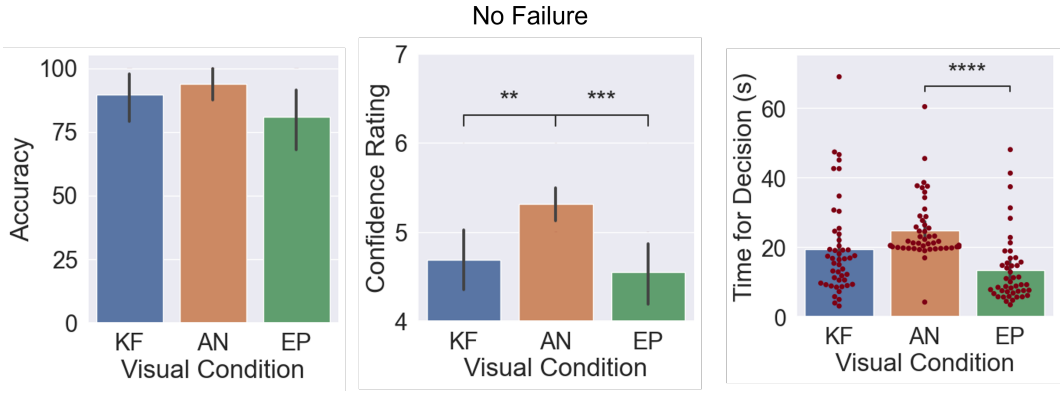


Figure 6.3: Accuracy Score, Confidence rating, and Time to Evaluate of No Failure (**: 1.00e-3 ¡ p ¡= 1.00e-2) (***: 1.00e-4 ¡ p ¡= 1.00e-3) (****: p ¡= 1.00e-04)

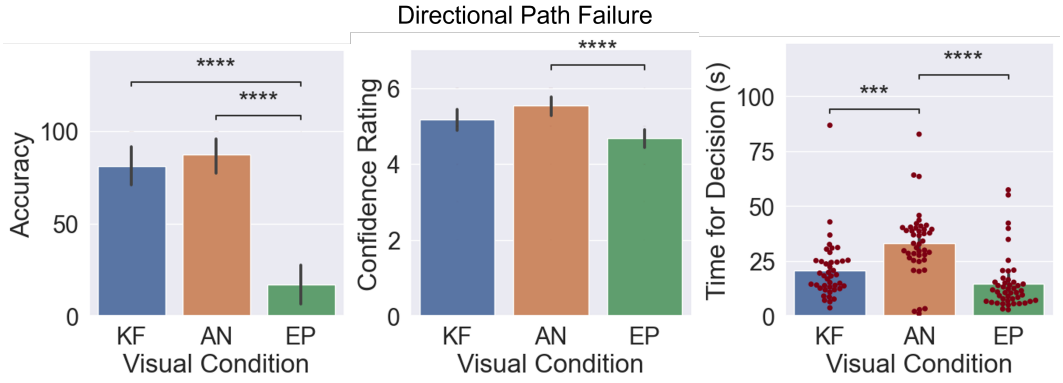


Figure 6.4: Accuracy Score, Confidence rating, and Time to Evaluate of Directional Path Failure (**: 1.00e-3 ¡ p ¡= 1.00e-2) (***: 1.00e-4 ¡ p ¡= 1.00e-3) (****: p ¡= 1.00e-04)

We analyzed the results of users evaluating a plan that was considered direct without any failures. We observed that the lowest score to be *End Points* (EP) with a value above 80% with no significant differences across visualizations. However, we did observe a significant main effect when users reported their confidence in their evaluation $F(2, 140) = 7.02$, $p = .001$. Post Hoc tests revealed *Animated* visuals boost users' confidence compared to *Key Frames* and *End Points* with p=5.36e-3 and p=4.34e-4 respectively. Consequently, when analyzing the average time duration determining the outcome of a robotic plan, we see a significant main affect *F(2,*

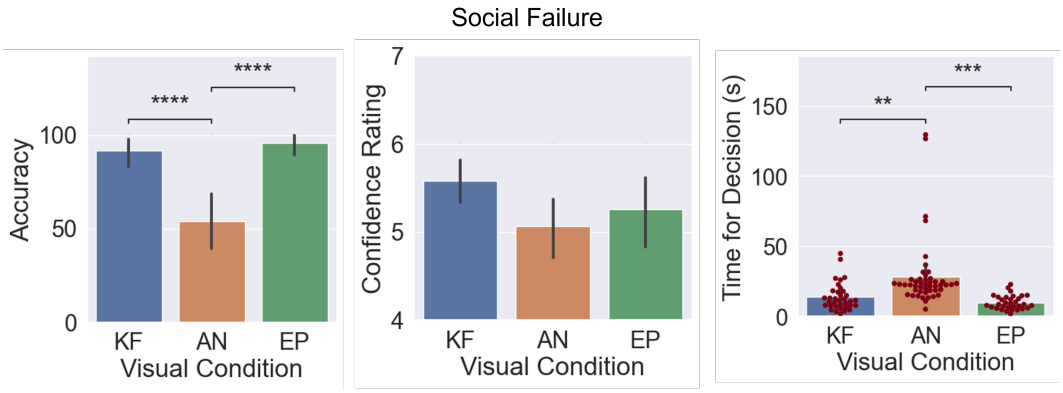


Figure 6.5: Accuracy Score, Confidence rating, and Time to Evaluate of Social Failure Plan (**: 1.00e-3 ¡ p ¡= 1.00e-2) (***: 1.00e-4 ¡ p ¡= 1.00e-3) (****: p ¡= 1.00e-04)

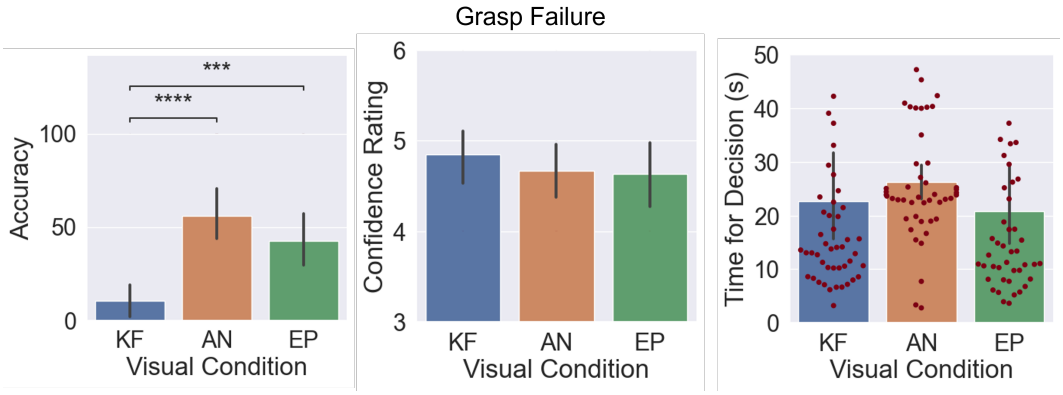


Figure 6.6: Accuracy Score, Confidence rating, and Time to Evaluate of Grasp Failure. (***: 1.00e-4 ¡ p ¡= 1.00e-3) (****: p ¡= 1.00e-4)

137) = 12.1, p ¡.001 which showed *Animated* taking additional time for evaluation compared to *End Points* (p =1.09e-7). (See Fig. 6.3)

We analyzed the results of users evaluating a plan that contained a directional path failure that guaranteed a spill. We observed that users with EP scored the lowest with an average of 17% while AN and KF both averaged above 80%. We found a significant main effect *F(2, 140) = 52.5, p ¡.001.* Post Hoc tests revealed that EP significantly decreased user's accuracy in determining a plan with a directional path failure compared to AN and KF (p¡.001). Confidence ratings revealed that participants on average were more confident compared to KF and EP with a significant main effect *F(2, 140) = 9.92, p ¡.001.* Post Hoc tests revealed that AN

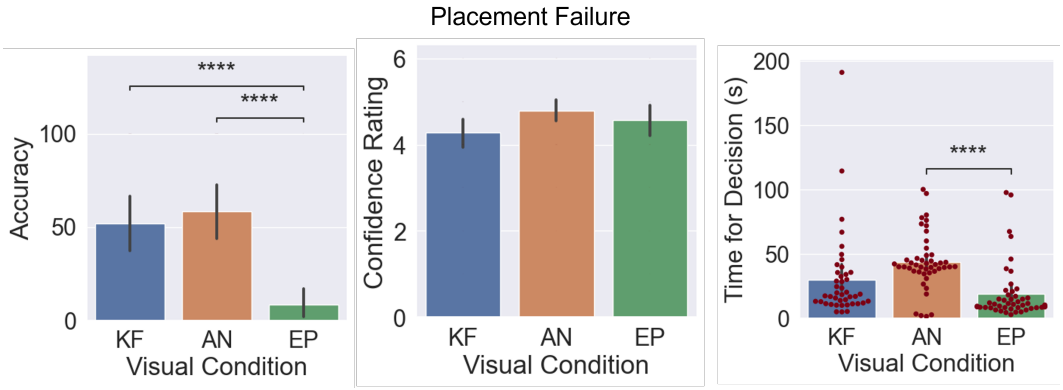


Figure 6.7: Accuracy Score, Confidence rating, and Time to Evaluate of Placement Failure. (***: 1.00e-4 ¡ p ¡= 1.00e-3) (****: p ¡= 1.00e-4)

significantly improved user confidence compared to EP (p¡.001). Elapsed Time measurements showed a similar pattern where AN takes almost 2x the time compared to KF and EP with a significant main effect *F(2, 128) = 20.2, p ¡.001.* Post Hoc tests revealed AN visuals required more time to process the visual information compared to EF and KF (p¡.001).

Finally we analyzed the results of users evaluating a plan that contained a socially unacceptable failure in which the robot places its gripper fingers within the cup in order to complete its task. The lowest score was AN with an average of 54% while KF and EP averaged above 90%, and we found a significant main effect among the conditions *F(2, 140) = 20.1, p ¡.001* Post hoc tests showed that AN significantly decreases users' ability to determine outcome of plans that contain social failure. Confidence ratings showed that participants all felt "somewhat confident" in there decision, but no significant differences were found. Elapsed time results showed a significant main effect *F(2, 111) = 13.6, p ¡.001* with post hoc tests showing AN requiring additional time compared to both KF and EP (p¡.001).

We analyzed the results of users evaluating a plan that contained a grasp failure where the robot cannot securely grasp the cup ultimately failing the task. We found that participants with KF scored the lowest with an average of 11%, AN scored the highest with 56% while KF scored 43%. We found a significant main effect *F(2, 139) = 13.0, p ¡.001* Post Hoc test revealed that KF significantly decreases

users' ability to detect grasp failure compared to AN (p¡.001) and EP (p=.002) respectively. Users' confidence ratings averaged around "somewhat confident" but showed no significant differences among visual groups. Elapsed time showed that all groups required at least 20 seconds on average to make a decision however we found no significant main effect.

We analyzed the results of users evaluating a plan that contained a placement failure where the robot cannot complete the task as the cup collides with the table ultimately spilling and found a significant main effect *F(2, 140) = 17.8, p ¡.001.* Users with EP scored the lowest with an average of 8% while KF and AN averaged around 50%. Post Hoc tests shows that EP significantly decreases users' accuracy in determining placement failures compared to AN and KF (p¡.001). Confidence ratings showed that users in all groups averaged along "uncertain/neutral" with no significant differences. Elapsed time results showed a similar profile in amount of time required for participants to determine the outcome of the plan among the visual conditions. A significant main effect was detected *F(2, 133) = 10.9, p ¡.001* among visual conditions, and post hot tests showed AN required additional time by users compared to EP (p¡.001)

We evaluated the NASA-TLX questionnaire which is the self reported cognitive load exam and plotted the results by the 6 subjective subscales (see Figure 6.8). With the task of evaluating robotic plans via augmented reality, we noted relatively higher ratings for Mental, Effort, and Performance, while the *Physical* subscale averaged the lowest ratings. The Mental demand subscale showed a significant main effect *F(2, 143) = 10.2, p ¡.001* with *Animated* ratings averaging the lowest compared to the other visual conditions. Post-hoc comparisons using Tukey's Honestly Significant Difference (THSD) revealed that *Animated* showed lower mental demand compared to both *Key Frames* (p = 3.02e-4) and *End-Points* (p = 7.32e-3). The Effort demand subscale showed a significant main effect *F(2, 143) = 5.76, p =.004* with *Animated* once again rated on average lower compared to the other visual conditions. Post-hoc comparisons using THSD revealed that *Animated* showed lower effort demand compared to both *Key Frames* ($p = .012$) and *End-Points* ($p = 0.01$).

The Performance demand subscale showed a relatively high rating around 6/10 but no significance difference was found among visual condition.

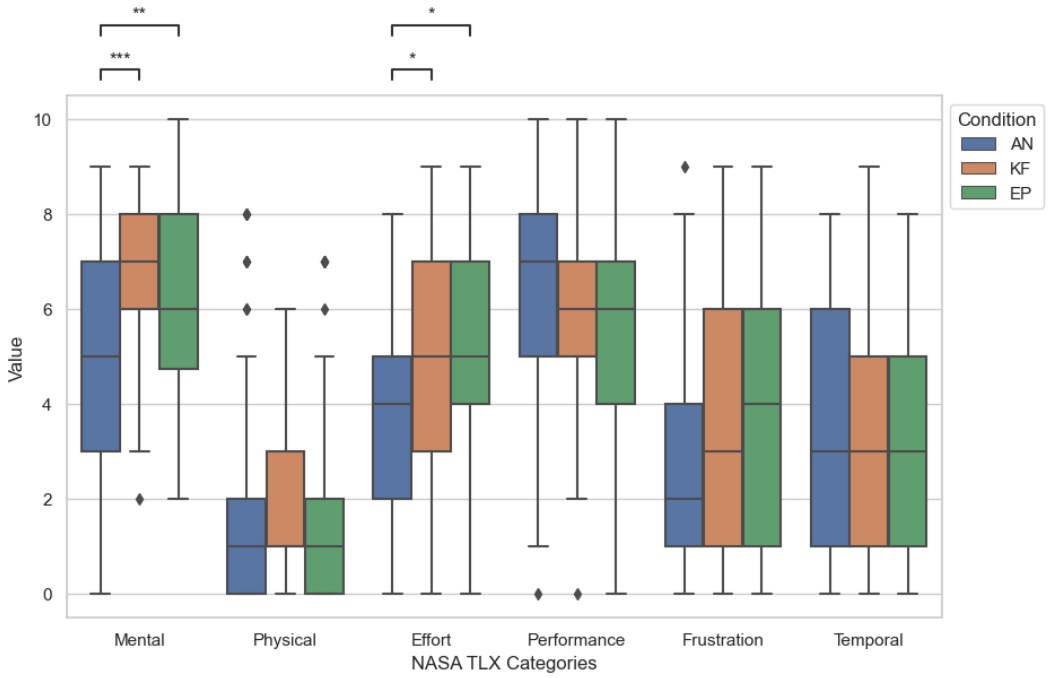


Figure 6.8: Results from the NASA TLX survey. Plot shows the average response value [0-10] for each NASA-TLX category. AN = Animated, KF = Key Frames, EP = End Points. *: 1.00e-02 ¡ p ¡= 5.00e-02, **: 1.00e-03 ¡ p ¡= 1.00e-02, ***: 1.00e-04 ¡ p ¡= 1.00e-03

Additionally, we analyzed the Detector Response Test (DRT) to measure cognitive load, where the higher the score, the less cognitive load a user experiences (see Figure 6.9). We found a marginal main effect by visual condition, $F(2, 130) = 3.08$, $p < .049$. Post-hoc comparisons using THSD revealed only that users with *Animated* visuals had a lower cognitive demand compared to *Key Frames* ($p = 1.312\text{e-}2$).

We ran a Sample Size calculation to ensure we recruited an appropriate number of participants. With a target Power of 0.80, a significance level $\alpha = 0.05$, and the effect size $\delta = 2.77$ to determine the number of participants we needed to run was approximately 6 per group, thus a total of 18, which is well below the total number we recruited.

We analyzed the percentage of time spent for each visual condition during the

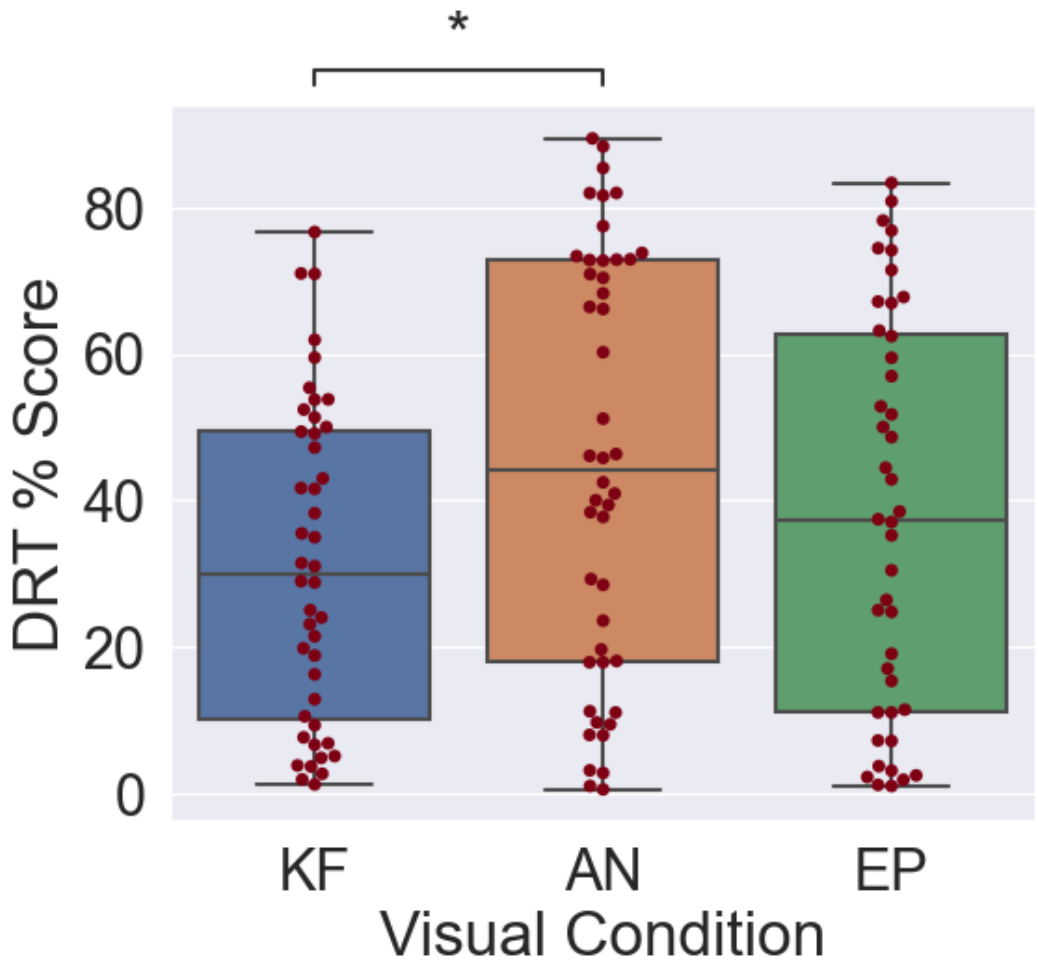


Figure 6.9: Average Detector-Response Test (DRT) scores across visual conditions. Bars indicate standard error. * indicates a marginal difference between *Key Frames* and *Animated*, ($p = .049$)

open-condition in which participants continued to evaluate plans with the freedom now to select whichever visual condition they please. Figure 6.11) shows a strong preference towards using the *Animated* condition with a percent average of 75.6% (SD = 34.2) resulting in a significant main effect *F(2, 138) = 85.0, p ¡.001.* Post-hoc comparisons using THSD revealed *Animated* conditioned leading over *Key Frames* ($p = 8.60e\text{-}14$) and *End Points* ($p = 5.80e\text{-}21$) respectively. Participants spent the least amount of time with the *End Points* conditions with average of 6.67%.

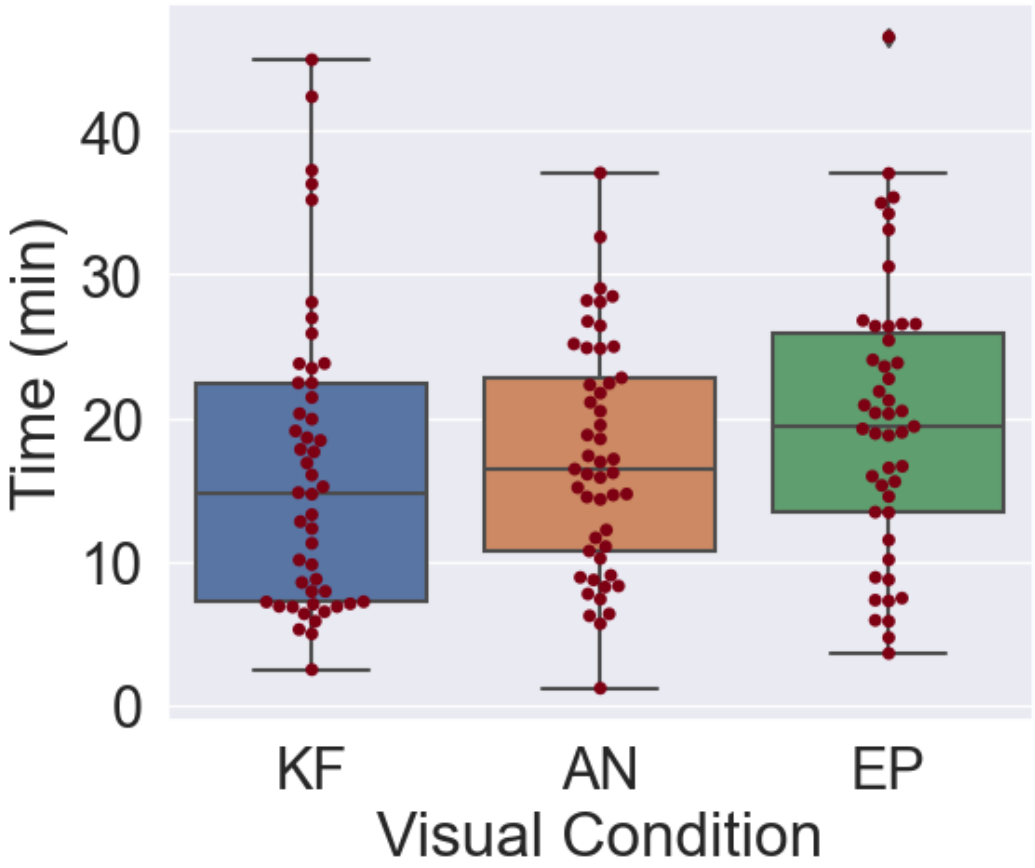


Figure 6.10: Average time in minutes to complete trajectory evaluation task across conditions.

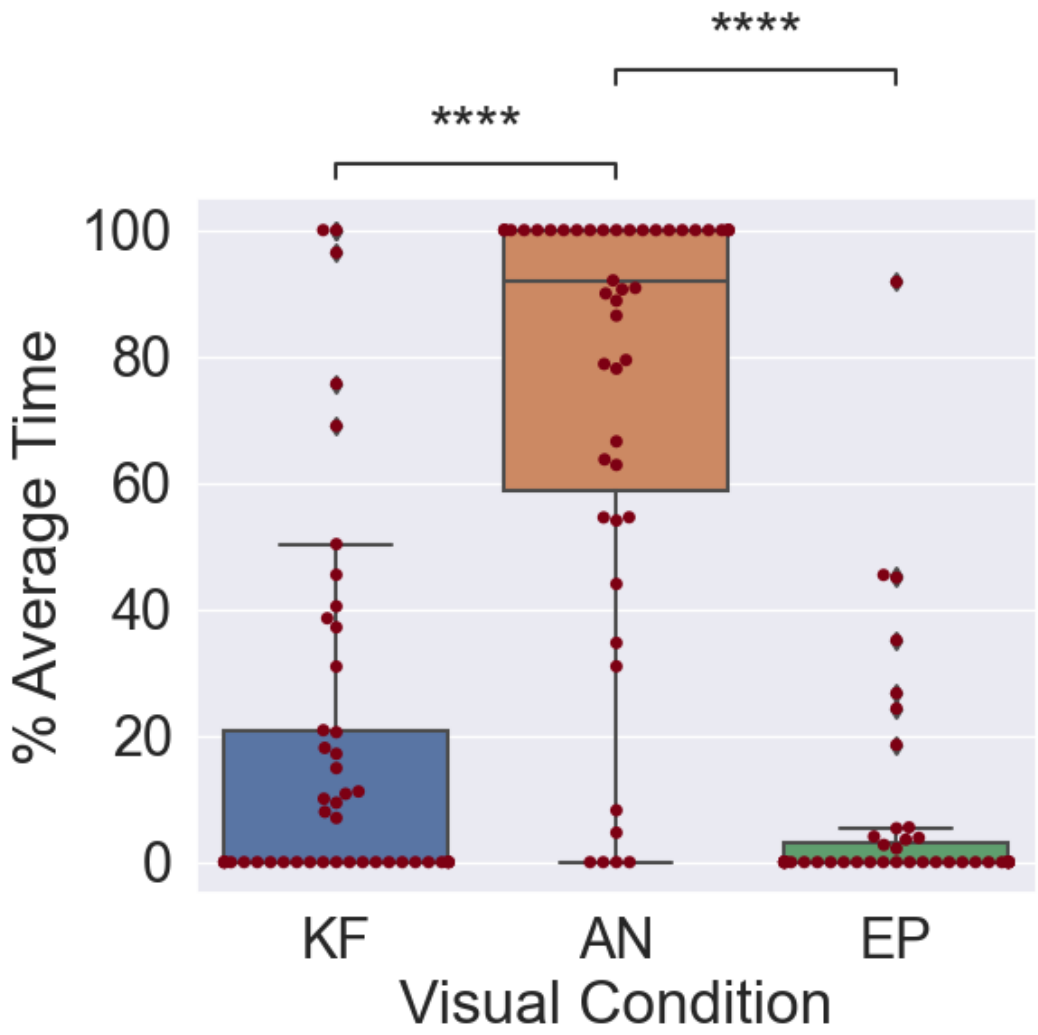


Figure 6.11: Percent Average of time using each visual condition to complete trajectory evaluation across conditions. (****: p ¡= 1.00e-04)

6.5 Discussions

Looking into the results of participants evaluating the outcome of various failure cases against our three visual conditions: *End Points* (EP), *Key Frames* (KF), *Animated* (AN), we find **H1** is partially supported with both and objective and subjective data.

For the direct plan without any failures, we see that all three visuals performed well in terms of *accuracy*. Both AN and KF revealed no unexpected maneuvers and EP revealed no issues with grasp pose and cup placement. EP however does not provide information in between frames and therefore presented no reason for a failure to occur. We see *Elapsed Time* is greater for AN because participants would need to watch the trajectory playback to gather information regarding the placement behavior unlike EP where they are given grasp and placement information and therefore it wouldn't take much time to interpret. KF makes sense that it falls between AN and EP because some time is required to interpret the flow of the plan among the visual clutter. AN would make sense that it ranks greater in confidence ratings compared to EP because information between the start and end of the plan are revealed unlike EP where is it unclear of what happens in the middle of the plan:

Looking into the directional path failure plan we see that EP has the disadvantage. In this plan, the robot makes a poor maneuver that inverts the cup before realigning itself upright for cup placement. It was expected to see that EP would not score as well because AN and KF would capture the inverted portion of the plan and thus indicate a spill. In the perspective of EP, it would appear as normal and we see a confidence similar to the average in the no failure case.

> **[P:12]** *"[End Points] visualization was immediately useful to get a good sense from beginning to end of the poses, provided that the movement wasn't too complex. "*
>
> **[P:2]** *"I found the organization of the Gripper Visualization to be nice since the screen wasn't too cluttered, but, because there wasn't a lot of in-*

formation with it, I had a harder time deciding if the robot would succeed in its task."

A participant reported the uncertainty that surfaces once going from AN to EP:

[P:19] *"[...] it became very apparent the level of unknown it[End Points] introduced after using the animations."*

The grasp failure condition involved the robot aiming to grasp the cup with its gripper parallel with the table but ultimately missing the cup because of the robot's shallow reach. The results indicated some difficulty in determining the plan's outcome as the averages for all conditions were below 60% and Users' confidence ratings leveled below "somewhat confident". KF could have been overwhelming with the number of overlapping visuals making it difficult to discern the gripper pose relative to the target container. EP may have performed better due to the reduced clutter within the participants' field of view as a few participants mentioned:

[P:6] *"[...] ¡Key Frames¿ However, I found the number of gripper images overwhelming, and it was sometimes hard to see where the cup was."*

[P:33] *"[...]¡Key Frames¿ it wasn't always clear exactly how it would grip the cup."*

[P:30] *"The key frame was easier to understand but sometimes I got lost trying to figure out where to look next for the following gripper pose"*

AN may have been more clear with the rendered gripper reaching above the cup. Although it may have not been distinct enough with the gap and some users may have made the assumption that it was good enough as some participants have pointed out the frustration with waiting for the playback to loop again:

[P:6] *"I liked this visualization ¡Animation¿ the least because every time I wanted to reference back to the trajectory, I had to watch the entire trajectory play out, which was annoying because I was also trying to pay*

attention to the secondary dot task. I wanted to see the key points in the trajectory as quickly as possible."

When analyzing the Placement failure plan where the robot moves its gripper closer to the table and ultimately forcing the grasped cup to collide with the tray trying to place it. All groups were expected to achieve high accuracy similar to the no failure plan as we reasoned that the indicator for the failure of the plan would have been the placement position showing the virtual gripper and cup penetrating through the physical tray. Similar to the grasp failure case, the rendering of visuals may not have been clear enough for some participants. AN and KF may have revealed directional path information that was not conveyed in EP which could have been an indicator with the visuals rendering too close to the table.

[P:8] *"¡Animation¿ Best overall but due to the noise in rendering was slightly less reliable to keep my trust high on gripper success relative to end point of arm"*

This rendering issue could point out a hardware limitation with the quality of the visuals interacting with physical objects. One participant however hinted at rendering an additional surface indicating table top detection:

[P:9] *"[...]] I would have liked to see the table as well to know for sure if the mug would ever hit the table."*

The social failure plan was direct with no inverted maneuvers; however, the robot motioned to grasp the cup with its gripper placed within the cup. In turn, we expected to see all groups to conclude the plan a failure. Surprisingly, the results showed that AN scored the lowest with almost half the participants determining the plan was acceptable. In contrast, EP and KF scored well above 90%, so it remains unclear as to why AN predicted poorly. We want to note that the the robot does in fact complete the task of placing the cup onto the participants tray, so it may be possible that seeing the completed trajectory may have users believe the plan was

acceptable.

We originally believed that AN gave more information about the plan that is not captured within KF or EP such as the robot's speed. This factor could be a reason as some participants reported that they factored the visual speed into their decisions even though speed was maintained and constant across all plans:

> **[P:20]** *"The animation was the easiest for me since it demonstrated the speed and angle of the arm."*

However, this logic was proven false as the reported reasoning for the failure for this plan showed nearly 75% of participants believed that the plan failed due to "Robot Grasps Cup Poorly" and 0% reported that the "robot moved either too fast/slow".

Based on these findings, we found **H1a** supported while **H1b** is partially supported in the case of accurately detecting failures involving poor directional trajectories. We found **H1c** support while **H1d** was partially supported except in cases in determining failures in plans with poor directional trajectories.

We understood that there may be trade-offs between robotic trajectory plans represented as either static visuals or animated playback. We reasoned that participants may have to spend some time to think and remember parts of an *Animated* plan to make a good judgement unlike the static visual *Key Frames* and *End Points*. The static visuals may not provide enough information (i.e., speed) for participants to make a fully informed decision, and as a result, could require additional reasoning on their end increasing mental demand. When analyzing the open condition portion of the study, we see a clear preference towards using *Animated* to evaluate robotic plans supporting **H2**. The open condition also revealed a switching behavior in which a participant would begin with one condition such as AN and observe the playback for the first part then quickly switch visuals to KF as a way to quickly gather the remaining information they require to make a decision. We noted approximately 45% of participants have been noted to switch between the visual conditions to gather more information to save time.

> **[P:15]** *"This is great ¡animated¿ but slow; swapping between the gripper*

visualization and this was nice when I needed more detail."

This act of swapping out visuals could also indicate the benefit of a mix condition of the visuals as one participant point out:

> [**P:8**] *"Combining end effector static with arm animation would be really interesting"*

Based on the NASA-TLX results, all visual conditions showed a large cognitive demand in the category of performance. This high rating could be due the nature of the study with participants ultimately predicting the outcome of a trajectory. We also saw relatively large ratings in the categories of mental and effort; however, AN showed to significantly decrease the demand levels compared to the key frames and end points. AN was reported to provide the most information and thus reduce any ambiguity:

> [**P:22**] *"This is the best option of the three because it takes the guess work out and provides the most amount of information but takes the most amount of time since you have to wait until the animation is over in order to continue."*

Physical demand rated the lowest as participants were only evaluating plans while mostly standing in one location. The DRT score offered a more objective cognitive load measure and revealed KF to significantly increase cognitive load compared to the other visual condition. Based on these findings, **H3** is not supported. An additional note to include in this report is that several participants mentioned that the DRT task was the more frustrating which may have contributed to the participants' overall cognitive load suggesting that secondary tasks are not recommended during trajectory evaluations.

> [**P:13**] *"[...], it was also more mentally demanding especially when clicking the red button."*

6.6 Limitations & Future Work

While our study successfully demonstrated several instances of robotic pick-and-place failures under three different visual conditions, it is important to acknowledge certain limitations.

Firstly, the use of the Detector-Response Test (DRT) was adapted from its original purpose of measuring cognitive load in the context of operating a motor vehicle. Although this adaptation followed a similar setup to a previous study utilizing a virtual reality headset, it may not fully capture the cognitive demands specific to interacting with robotic systems in augmented reality environments.

Secondly, the AR device (Hololens2) utilized in our study occasionally experienced lag, particularly noticeable in the Key Frames condition. This lag may have been attributed to the heavy processing load imposed by continuously tracking and rendering a large number of virtual objects. This issue could have influenced participants to switch from the Key Frames condition to another visual condition. An upgraded AR device with enhanced processing capabilities could potentially mitigate this issue and provide a smoother user experience.

Additionally, the environmental setup in our study simulated a controlled scenario with target containers placed in a predetermined location. In real-world settings, target containers would exhibit varying geometries and may be situated in diverse locations, posing additional challenges for robotic pick-and-place tasks. Furthermore, while we curated specific types of failures for this study, it is acknowledged that this selection may not encompass the entire spectrum of potential failures in real-world scenarios. Future studies could explore a broader range of delivery tasks and failure scenarios to provide a more comprehensive understanding of the challenges faced by robotic systems.

In terms of future work, one promising avenue involves investigating user control over specific parts of the robot's planned trajectory that users perceive as potential points of failure. This approach aligns with the findings of systematic comparisons presented in prior research [21], and exploring such differences when

presenting multiple trajectory plans through AR could offer valuable insights into user preferences and decision-making processes.

6.7 Conclusions

The objective of this research was to determine the adequacy of visualized information in augmented reality (AR) for participants to identify failures in a pick-and-place plan involving robotic manipulation. Through a human-participant study, we investigated users' accuracy in discerning the outcomes of robotic plans for delivering a cup of water to a user's tray table when observed through augmented reality. We introduced plans designed to fail in four distinct ways: grasp failure, directional path failure, social failure, and placement failure. Our study manipulated the independent variable of visual condition in AR, which took three forms with varying levels of information: End Points, Key Frames, and Animated.
Through the completion of this study, several key findings emerged:

- Both Key Frames and End Points offer quick trajectory information, but they struggle to convey the robot's speed effectively.
- Users are better able to detect failures in robotic plans related to poor path directions when using Key Frames and Animated representations compared to End Points, albeit at the cost of increased time required for analysis.
- Participants generally prefer Animated representations of robotic plans, particularly in the context of pick-and-place tasks involving fluid manipulation, if given the choice. A minority of participants demonstrated a tendency to switch between visual representations as a validation strategy.
- Detector-Response Test (DRT) scores indicate that Key Frames are more overwhelming for the evaluation of robotic plans compared to End Points and Animated representations, particularly when participants are engaged in a secondary task.

These findings hold implications for future research and development in mixed reality applications for robotic object manipulation. While augmented reality serves as a valuable tool for visualizing robotic information in an intuitive manner, the manner in which information is rendered is crucial. Different modes of visualization have the potential to enhance users' abilities to provide corrective information for robots and empower users to mitigate undesired outcomes.

Chapter 7

Future Work

A primary objective of the research outlined in this dissertation is to delve deeper into the potential of Augmented Reality (AR) in improving the collaboration between robots and individuals, regardless of their expertise in robotics. While the studies presented in this work represent incremental progress in the Human-Robot Interaction community, there exist several unexplored avenues within robotics that could benefit from AR in future work. One pivotal area is the exploration of Augmented Reality for Enhanced Robot Training. Furthermore, teaching by demonstration is a prevalent approach in training intelligent robots for personalized tasks. While training typically occurs in controlled settings like research labs, the challenge lies in the subsequent deployment of trained robots in real-world environments, which necessitates retraining due to perceptual variations.

For instance, the study in Chapter 5 can be extrapolated to a reinforcement learning context, where the objective is to facilitate a robot in learning to generalize a policy for successful pouring from various source containers into a target container without spillage. In essence, this leads to the general question: *how can we employ AR to visualize a robot's learning policy to expedite human-guided training?*

Based in this context, we can ask further questions:

7.0.1 What information of a robotic reinforcement learning (RL) problem do people benefit the most when visualized in AR?

This entails identifying the most beneficial information from a robotic reinforcement learning problem when visualized in AR. This could encompass the spatial distribution of observations and desired actions at each state. As discussed in Chapter 4, non-spatial data such as cognitive decisions and diagnostic information may not be useful when visualized; therefore, we can hypothesize that RL parameters such as weights and hyper-parameters may not provide any benefits.

7.0.2 Enable modifications of a robotic scene to better improve the robot's learning process based on visualizations determined in the first study?

Exploring whether individuals can effectively modify the robotic scene to enhance the robot's learning process based on visualizations derived from the initial study. This might involve repositioning the robot, sequencing specific actions, or correcting the robot's actions or observations in a given state based on the information conveyed by the robot. We describe this feature in more detail in the next section 7.1.

7.0.3 Development of a dynamic visualization algorithm that is adaptive in visualizing concise information for training a robot in context of a reinforcement learning problem

With the complexity of training a physical robot, people may require various types of the robot's information in order to properly train the robot. An adaptive visual interface may be useful to avoid any overwhelming visual clutter leading to a loading cognitive burden.

7.1 TRAinAR[§§]

Teaching by demonstration is a common approach to train intelligent robots for personalized tasks, where a teacher will manually control (teleoperation, kinesthetic programming) a robot to complete a task in order to speed up or improve the robot's learning process. Training a robot often takes place in a controlled setting such as a research lab by roboticists; however, trained robots are expected to fail when deployed to new environments (i.e., homes, public spaces) due to changes in perceptions that differ from the initial trained setting and thus will require retraining.

Users unfamiliar with robots are challenged when needing to teach a robot a new personalized task because of the lack of communication of what has been learned resulting in the user making poor mental models of the robot; therefore, it is crucial that a teacher receives adequate feedback from a robot in the teaching process. Augmented reality (AR) technology has shown to be an alternative medium in conveying complex robotic information through simplified visualizations [118, 35, 33, 34], and therefore, we hypothesize that visualizing the robotic learning process, which is often a "black box" to many individuals, may provide useful information for a user to train a robot effectively. Specifically, we ask the questions, "Can humans become better teachers to robots with AR technology?" and, "What information visualized do users benefit the most when training and teaching a robot?"

To address these questions, we propose a human-participant study where users are instructed to improve a robot's task with the help of an AR device. We developed ***TRAinAR*** (Teaching Robots Actions in Augmented Reality) which aims to enable a user to 1) quickly prototype a 3D virtual training environment with constraints that reflect the real-world, 2) visualize the robot's training data and learn-

[§§]**This chapter is based on the following papers:** Andre Cleaver, and Jivko Sinapov. Helping Humans Become Better Teachers for Robots with Augmented Reality. Accepted to the ACM/IEEE International Conference on Human-Robot Interaction (Workshop VAM-HRI 2023) and Andre Cleaver, and Jivko Sinapov. Demonstrating TRAinAR: An Augmented Reality Tool that Helps Humans Teach Robots. Accepted to the ACM/IEEE International Conference on Human-Robot Interaction (HRI 2023)

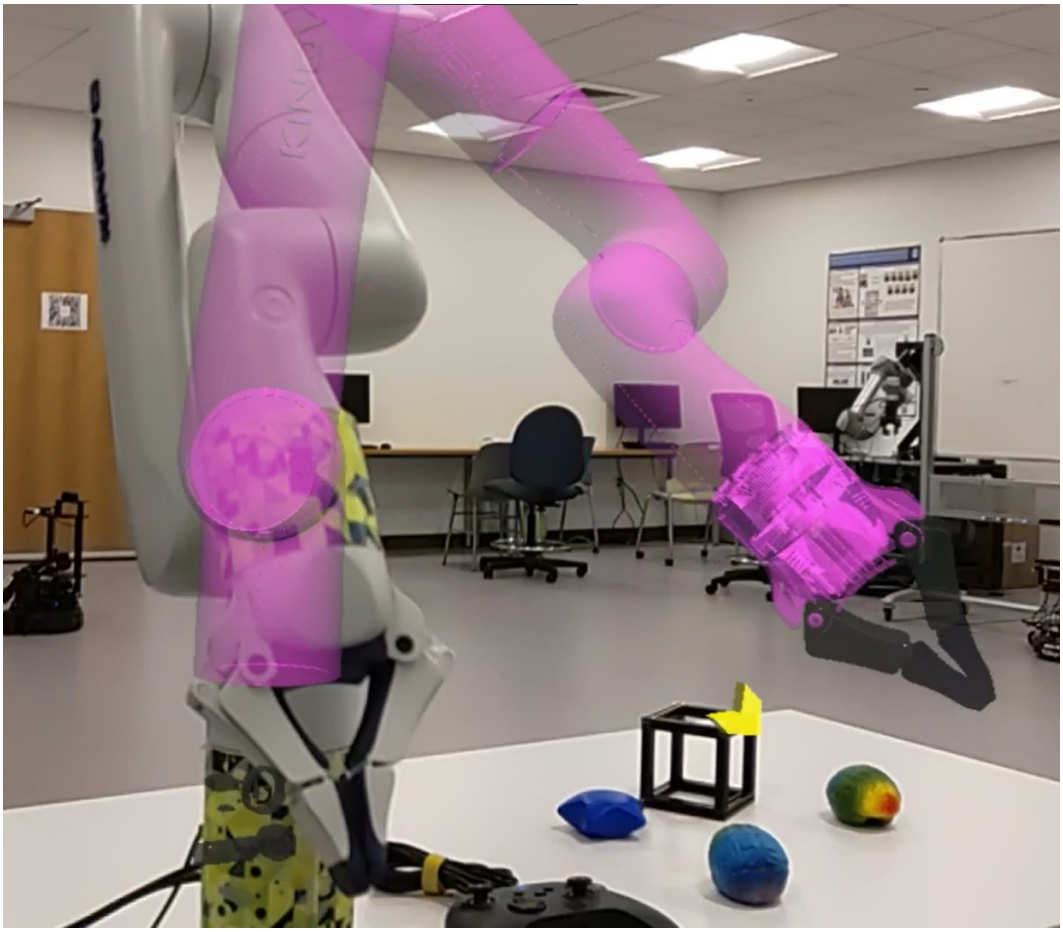


Figure 7.1: Kinova Gen3 Lite robot with augmented digital twin conveying motion intent (purple arm) with target goal pose (yellow arrow).

ing parameters, and 3) animate a robot's future actions before execution. Results from the proposed user study will determine *TRAinAR* as a viable tool for robotic sim2real training and provide insights into visualized robot data for improving the training process. In a technical demonstration, our application enabled a robot to learn how to navigate its gripper toward a target object while implicitly avoiding obstacles.

7.2 Related Work

Explainable robotics and AI is an emerging field that aims to address the "black box" issue regarding decision making by advanced robotics or artificial intelligent (AI) systems that control a robot [76]. Reinforcement learning (RL) and machine

learning algorithms also fall under this category where the decision pipelines are not transparent making it difficult for humans to put their trust with a robot [147]. Additionally, the lack of an explanation behind a robot's decisions can result in mismatched mental models between a user and the robot ultimately leading to undesired outcomes [166, 70, 43]. As a result, studies have looked into robotic systems that convey their decision process to humans. [39] evaluated robotic systems in an RL setting through a goal-driven approach with robot decisions explained in terms of probabilities of success for easy interpretation. [4] demonstrated virtual robots that were perceived as more lively and human-like when explaining gaming strategies out loud. [175] proposed an experimental framework combining AR with explainable AI with a virtual robot. The AR system visualized the perceptional beliefs of the robot as the user interacts with virtual objects that were part of a daily activity.

More notable works explored training a virtual agent in a virtual environment using neural networks and later transferring the knowledge to a real robot. [52] and [158] both trained robotic systems using humans demonstrating grasping techniques in virtual reality that later transferred to a physical robot. [190] similarly trained a robotic system across several types of object manipulation tasks with virtual reality teleoperation. [105] demonstrated constrained learning applied to a robotic end effector with augmented reality to improve humans teaching robots various skills. Inspired by the *ARC-LfD* project by [104], we aim to create *TRAinAR* as a tool to quickly prototype complex environments in virtual environments with a similar form of augmented constraints. In our proposed human-participant study, participants will train a robotic arm manipulator given the capabilities to modify the virtual training environment as well as access to the robot's existing knowledge of the task.

7.3 Methodology

To determine *TRAinAR* as a viable training tool, we propose a between human-participant study. Participants are assigned to either the test group or control

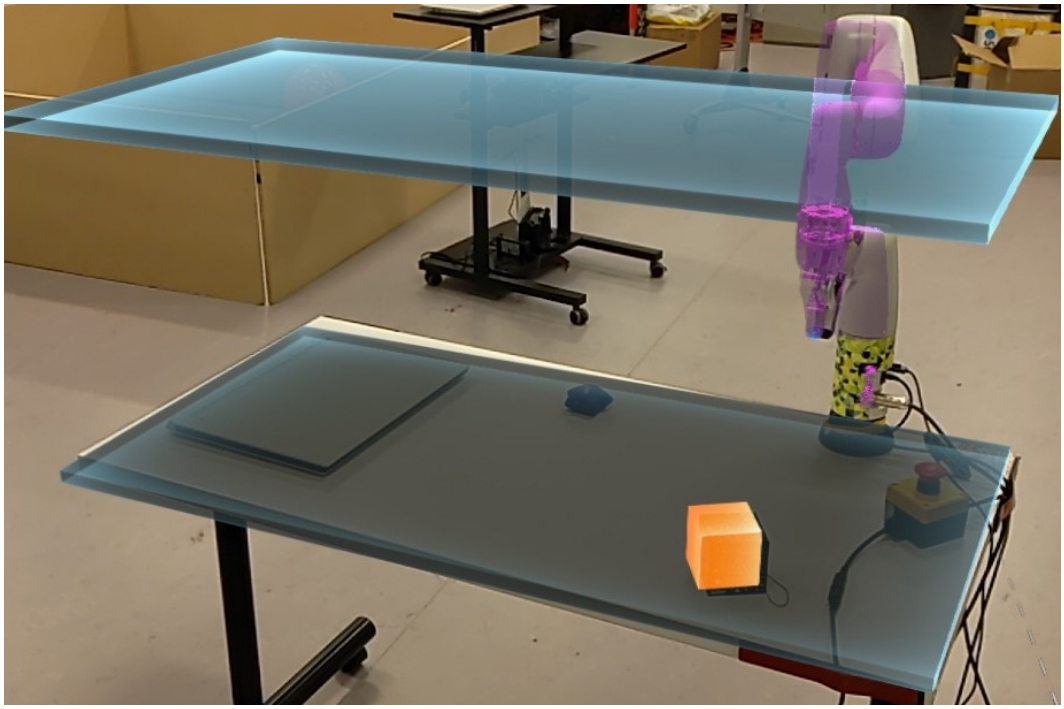


Figure 7.2: Kinova Gen3 Lite robot with virtual boundary planes (ground and ceiling) that restricts both the robot's movement as well as the exploration space. Users can physically interact with both planes to constrain the environment to their preference.

group. The difference between the groups is how the information is visualized: on a standard computer monitor or through an augmented reality device. The control group will only be interacting with a laptop computer, and the test group will interact through the augmented reality (AR) device.

For each participant, the entire session is split into three parts: a pre-questionnaire with tutorial, robot training, and a post questionnaire. The tutorial will focus on how to use *TRAinAR* along with a tutorial on how to control the robotic arm manipulator via game controller. Participants in the test group will be fitted with the Hololens2 and shown how to interact with the virtual environment through hand gestures. Participants in the control group will only use a standard laptop computer with a mouse. Both groups will proceed to complete the following tasks:

7.3.1 Task Descriptions

7.3.1.1 Modeling the Environment

In this task, the participant will modify a virtual training environment to constrain the robot to stay in its working zone and avoid collisions with other objects. The participant will set the constraints for the robot by repositioning the virtual objects that correspond to its real-world counterpart. For example, a participant will reposition a bottom boundary plane that corresponds to the physical table that was purposely misplaced. The participant in turn is expected to reposition the virtual plane to match the height of the physical table or superimpose the virtual plane over the physical table. Participants wearing the AR device will perform a pinch gesture to manually reposition the virtual plane, and the participants using the computer will use a mouse to click and drag the virtual plane to where they believe the plane should be placed. The changes to the virtual environment will be recorded and then recreated by the PI for later training.

7.3.1.2 Teaching by Demonstration

In this task, we will instruct the participant that the task the robot will learn is navigating its end-effector (gripper tip) to a target object which can be seen highlighted in orange in Figure 7.2.The participant will use a gaming controller to move the robot's arm to the object's location. The actions by the participant will be recorded, and the process of demonstration repeats for a set period of time.

7.3.1.3 Knowledge From the Robot's Training History

We include this task to investigate what a user can determine from the robot's training history data and how they would improve the robot's training process. For example, Figure 7.3 shows a history of an agent's spawned locations visualized in context after a training session. In a questionnaire, participants report sections where the robot has already explored and where the robot should explore next to expand its knowledge to improve its performance.

7.3.2 Hypotheses

H1: Users with AR will train the robot more confidently compared to users without AR.

H2: Users with AR feedback will require shorter training times to train the robot.

H3: Robots that have been train by users with AR feedback will have greater performance compared to robots that have been trained by users without AR feedback.

H4: Users that train the robot with AR will report lower cognitive load compared to users that train the robot without AR.

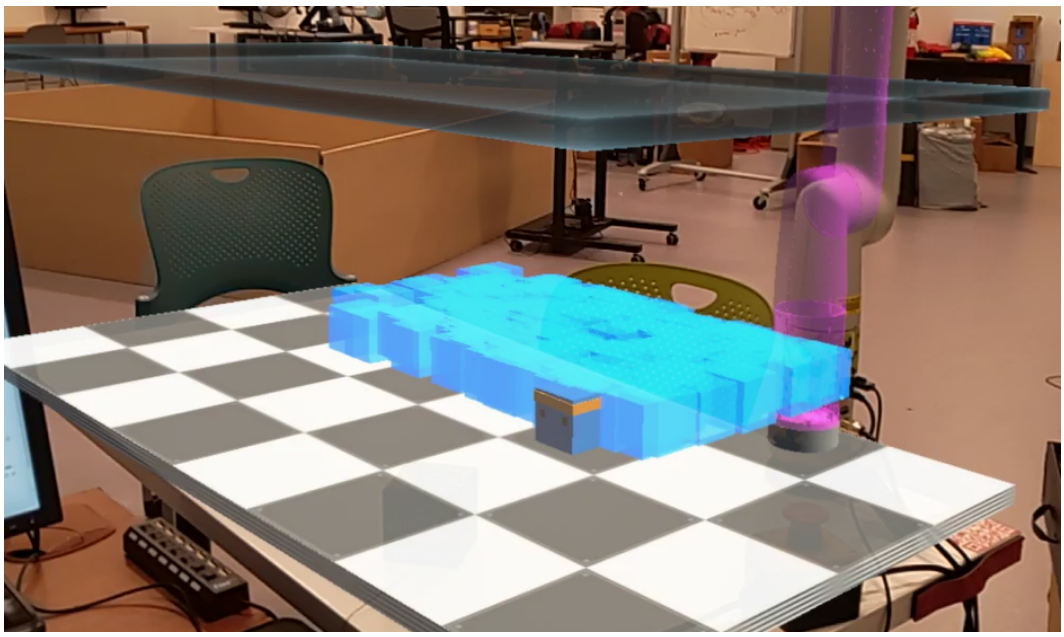


Figure 7.3: The user visualizes the training data history over the real-world environment. The agent here is represented as a blue cube which spawns in random locations for each training episode. Users with visualized training data can gain insights into the agent's learning performance.

7.3.3 Hardware & Software

We demonstrate *TRAinAR* using a 6 degrees of freedom *Kinova Gen3 Lite* robot controlled with *Robot Operating System* (ROS) [136] running on Ubuntu 18.04. Python and C++ scripts filtered the robotic data before delivery to the AR device for visualization. *TRAinAR* was developed with *Unity*[*] and deployed onto a (*Hololens2*) which rendered the robotic data over the physical environment.

Vuforia[†] tracked the robot's position and orientation through a cylindrical target-image fixed to the robot (see Figure 7.1). C# scripts control data exchange between the robot and AR device that occurred over a shared Wi-fi network using *ROS Sharp*[‡]. We utilized *Unity ML-Agents Toolkit* [90] to train a virtual agent that represented the real robot. We leveraged Microsoft's Mixed Reality Toolkit 2 (MRTK2) to handle user hand gesture tracking when manipulating with the virtual objects. Additional C# scripts logged the training history data such as the virtual agent's position throughout the training process.

7.3.4 The Virtual Training Environment

For simplicity, we aim to have participants train the robot to perform a classic move-to-target task; specifically, have the robot navigate its end-effector to a target object while avoiding obstacles. Unity's ML-agents toolkit by default uses proximal policy optimization to map the agent's observations to the best actions in a given state. In our case, our agent will learn to navigate from an initial starting position to a target object in 3D space. From Figure 7.2, the virtual planes are boundary constraints to limit the exploration space. A digital twin of the robotic arm is added to the virtual environment along with virtual objects to represent any additional obstacles that will be randomly placed in the real-world for the participants to account for. All virtual objects have rigid-body and collider components which enable users to interact with them. The training agent which represents the robotic end-effector

[*] https://unity.com/

[†] https://developer.vuforia.com/

[‡] https://github.com/siemens/ros-sharp

has a unique script that terminates a training episode once it makes collides with any object. Positive and negative rewards are assigned to the target object and obstacles respectively. We train the agent in continuous action space with 3 actions that correspond to x-y-z velocity inputs in Cartesian coordinate space. The agent's observation space includes the x-y-z positions of both the target object and the robot end-effector agent.

For teaching by demonstration, Unity's ML-Agents Toolkit has a *Demonstration Recorder* script that captures the user's actions for imitation learning. During the training phase of the study, participants will use a gaming controller to send actions to control the robot's end-effector. These actions will then be incorporated into the robot's learning algorithm to hopefully speed up the learning process.

7.3.5 Measures & Analysis

To answer our questions, we plan to gather a combination of objective and subjective measures. Both the Hololens2, laptop, and robot connect with ROS; therefore, we plan to record timestamped data for environment changes and user actions. We will consider overall demonstration times for both conditions, expecting shorter times for the test group (**H2**). The total training time in simulation for the agent is also expected to be shorter for the test group. The average reward collected in training is expected to be greater in the test group (**H3**). After completion of the tasks, participants will be administered a questionnaire to gauge how effective they believe they were in training the robot. User confidence will be recorded in a Likert scale which we anticipate greater ratings compared to the control group (**H1**). Measuring user cognitive load will be conducted with the NASA Task Load Index, expecting lower ratings for the test group (**H4**). Open-ended questions will prompt participants to describe helpful visuals.

7.4 Discussion

TRAinAR is proposed as a viable tool for improving the sim2real training process for Reinforcement Learning and robots. *TRAinAR* aims to enable users to learn more about robot's personalized task by visualizing the training environment and data history to shed light on any unexpected robot behaviors. As a result, users can modify a virtual training environment to improve the training process. In a technical demonstration [§], we show that a robot was able to learn to move towards a target object after tailoring a virtual training environment described in section 7.3.4.

[§] Video Demonstration

Chapter 8

Conclusions

We set out to explore further how Augmented Reality (AR) could enhance the interactions between humans and robots through the concept of a shared reality space. This space allows robots or intelligent agents to display their virtual world, encompassing their knowledge, intentions, and perceptions of the environment, which are typically concealed from humans. Conversely, humans can gain a clearer understanding of the robot and its perceptions, enabling them to provide feedback when necessary. The ultimate aim of a shared reality space is to bridge the communication gap between robots and humans by aligning their perceptions. This dissertation focused on exploring three key aspects of how this shared reality space facilitates human-robot collaboration.

First, Chapter 4 investigated the circumstances in which individuals prefer to visualize robot information using AR. We categorized robotic information into five distinct classes and designed a series of activities centered around each class. Participants equipped with AR devices, capable of visualizing robot information within the environment, predominantly found value in activities related to Object Detection and Safety.

Second, Chapter 5 demonstrated how a shared reality space aids in guiding manipulation behaviors, particularly in complex tasks like fluid manipulation. Pouring fluid into a cup remains challenging for robots due to the intricate nature of fluid behavior. Humans, with their innate understanding of physics, find such

tasks relatively simple. By visualizing the robot's pouring behavior and leveraging human intuition, we investigated the use of AR to guide successful pours with minimal spillage. Results from a human-participant study showed a reduction in spillage when using visualization tools, especially with container alignment during parameter control in angular rotation and horizontal positioning.

Third, Chapter 6 addressed the issue of visual clutter in AR applications, particularly concerning robot motion intent with arm manipulators. While providing users with more information about a robot's planned trajectory is essential for determining the success of an action, it often leads to visual clutter, hindering the user's view. We explored methods to reduce visual information, such as using simple points and lines to represent next goal locations. A human-participant study evaluated robot plans visualized in AR, focusing on a classic pick-and-place task with a cup of water. Results indicated that participants could identify directional maneuver failures with both animated and static key frame visuals.

Lastly, we determined for future work in Chapter 9 that a potential area that people can benefit with a shared reality space is robot learning. In general, the field of machine learning and reinforcement learning are often a "blackbox" to most non-experts and the challenges only rise when incorporating a physical robot to learn a task. Concepts such as *sim2real* attempt to shorten the learning process; however, training in simulation often do not match more complex real environments where the robot could potentially fail. As a potential solution, we introduced *TRAinAR*, an AR-based application that enables users to quickly prototype complex environments in simulation for robotic agent training and visualize the training history in context of the real environment. With such a tool, users can gain insights into the robot's learning process and could potentially better train the robot.

8.1 Lessons & Take Aways

Some lessons that emerged while developing the Augmented Reality systems used in the HRI studies to consider when applying to robotic systems in the future:

1. Ensuring the precision of digital twin models within the shared reality space is paramount. Insights gleaned from participants' responses to the open-ended inquiries on their encounters with augmented visuals, as described in Chapter 5 and Chapter 6 underscore the ramifications of flawed modeling. Inaccuracies can lead to participants receiving misleading information. Particularly in predicting the outcomes of robotic actions, participants may rely heavily on augmented visuals to guide their decisions. Consequently, any discrepancies in the geometric alignment of digital twins may yield outcomes characterized by diminished task accuracy despite participants' heightened confidence. For those considering overlaying models onto their real-world counterparts, employing exceptionally accurate models is strongly advised.

2. Take into account the diverse audience that will interact with the augmented visuals, including individuals who may be color blind or sensitive to bright lights mentioned in section 6.3.1. The properties assigned to virtual objects can be adjusted in various ways, such as modifying transparency, adjusting color schemes, and controlling brightness levels .

3. Tracking remains a persistent challenge for AR devices, as accurately determining the user's position and orientation is essential for proper rendering of augmented visuals. In our HRI studies, we addressed this by establishing an anchor point using a fiducial marker detected by the AR device's camera, enabling us to align robotic data with the robot's position and orientation. However, when the fiducial marker exits the AR device's view, there's a risk of augmented visuals drifting off, leading to misalignment (See section 4.5). While employing additional fiducial markers could mitigate this issue, alternative methods for ensuring precise augmented visual placement is worth exploration, although potentially requiring extra equipment.

4. Consideration of the distinct design space in which an AR system operates within an HRI setting is important, as outlined in 1.1.2. Visualizing robot data can be done in different ways, but understanding the context and situation

helps choose the right visualization style. Chapter 6 considers the design space in selecting the visualization conditions to address visual clutter.

8.2 Summary

Drawing upon the comprehensive exploration conducted in this dissertation, it is evident that Augmented Reality holds immense potential in enhancing human-robot interaction through the implementation of a shared reality space. Each chapter has contributed valuable insights into the practical applications and benefits of integrating AR technology into collaborative robotics scenarios.

By facilitating a deeper understanding between robots and humans, the shared reality space not only overcomes communication barriers but also fosters more effective collaboration and task completion. From visualizing robot information to guiding manipulation behaviors and addressing issues of visual density, the findings presented here underscore the significance of AR in bridging the gap between human and machine intelligence.

As we look ahead, the implications of this research extend far beyond the confines of this dissertation. Further development and extension of these concepts have the potential to revolutionize the field of robotics, bringing us closer to the ultimate goal of seamless coexistence and cooperation between robots and humans within shared environments.

In conclusion, this dissertation represents a significant step forward in leveraging Augmented Reality to enrich human-robot interaction. By continuing to refine and expand upon these ideas, we can propel the robotics community towards a future where collaborative robotics becomes not just a possibility, but a transformative reality

Bibliography

[1] Alexey Abramov et al. "Depth-supported real-time video segmentation with the Kinect". In: *2012 IEEE Workshop on the Applications of Computer Vision (WACV)*. ISSN: 1550-5790. Jan. 2012, pp. 457–464. DOI: 10.1109/WACV.2012.6163000.

[2] Pulkit Agrawal et al. "Learning to poke by poking: Experiential learning of intuitive physics". In: *Advances in neural information processing systems* 29 (2016).

[3] Nouf Matar Alzahrani. "Augmented reality: A systematic review of its benefits and challenges in e-learning contexts". In: *Applied Sciences* 10.16 (2020), p. 5660.

[4] Jakob Ambsdorf et al. "Explain yourself! Effects of Explanations in Human-Robot Interaction". In: *arXiv preprint arXiv:2204.04501* (2022).

[5] Stephanie Arevalo Arboleda et al. "Assisting manipulation and grasping in robot teleoperation with augmented reality visual cues". In: *Proceedings of the 2021 CHI conference on human factors in computing systems*. 2021, pp. 1–14.

[6] Kai O Arras and Dario Cerqui. "Localizing overlapping regions in the presence of false landmarks". In: *Robotics and Autonomous Systems* 51.3 (2005), pp. 195–207.

[7] Ronald Azuma and Gary Bishop. "Improving static and dynamic registration in an optical see-through HMD". In: *Proceedings of the 21st annual conference on Computer graphics and interactive techniques*. 1994, pp. 197–204.

[8] Ronald Azuma et al. "Recent Advances in Augmented Reality". en. In: *IEEE Computer Graphics and Applications* (2001), p. 14.

[9] Ronald T Azuma. "A survey of augmented reality". In: *Presence: teleoperators & virtual environments* 6.4 (1997), pp. 355–385.

[10] Renée Baillargeon. "Infants' understanding of the physical world". In: *Journal of the Neurological Sciences* 143.1-2 (1996), pp. 199–199.

[11] Renée Baillargeon. "The acquisition of physical knowledge in infancy: A summary in eight lessons". In: *Blackwell handbook of childhood cognitive development* (2002), pp. 47–83.

[12] Chris L Baker, Rebecca Saxe, and Joshua B Tenenbaum. "Action understanding as inverse planning". In: *Cognition* 113.3 (2009), pp. 329–349.

[13] Christopher Bates et al. "Humans predict liquid dynamics using probabilistic simulation." In: *CogSci*. 2015.

[14] James Baumeister et al. "Cognitive cost of using augmented reality displays". In: *IEEE transactions on visualization and computer graphics* 23.11 (2017), pp. 2378–2388.

[15] Cindy L Bethel and Robin R Murphy. "Nonverbal communication between humans and robots". In: *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 38.1 (2008), pp. 78–92.

[16] Mark Billinghurst and Hirokazu Kato. "Collaborative augmented reality". In: *Communications of the ACM* 45.7 (2002), pp. 64–70.

[17] Cynthia Breazeal. "Emotion and sociable humanoid robots". In: *International Journal of Human-Computer Studies* 59.1-2 (2003), pp. 119–155.

[18] Cynthia Breazeal et al. “Robots as intentional agents: Using neuroscientific methods to make robots appear more social”. In: *Cognitive Systems Research* 6.1 (2005), pp. 102–118.

[19] Susan E. Brennan and Eric A. Hulteen. “Interaction and feedback in a spoken language system: a theoretical framework”. en. In: *Knowledge-Based Sys.* Human-computer collaboration 8.2 (Apr. 1995), pp. 143–151. ISSN: 0950-7051. DOI: 10.1016/0950-7051(95)98376-H. (Visited on 10/25/2019).

[20] Andrew G Brooks and Cynthia Breazeal. “Working with robots and objects: Revisiting deictic reference for achieving spatial common ground”. In: *Proceedings of the 1st ACM SIGCHI/SIGART Con. on (HRI)*. 2006, pp. 297–304.

[21] Maya Cakmak et al. “Human preferences for robot-human hand-over configurations”. In: *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE. 2011, pp. 1986–1993.

[22] Josep Call and Michael Tomasello. “Does the chimpanzee have a theory of mind? 30 years later”. In: *Trends in cognitive sciences* 12.5 (2008), pp. 187–192.

[23] Elizabeth Cha et al. “A survey of nonverbal signaling methods for non-humanoid robots”. In: *Foundations and Trends® in Robotics* 6.4 (2018), pp. 211–323.

[24] Ravi Teja Chadalavada et al. “That’s on my mind! robot to human intention communication through on-board projection on shared floor space”. In: *2015 ECMR*. IEEE. 2015, pp. 1–6.

[25] Joyce Y. Chai et al. “Collaborative Effort Towards Common Ground in Situated Human-robot Dialogue”. In: *Proceedings of the 2014 ACM/IEEE Int. Con. on Human-robot Interaction*. HRI ’14. event-place: Bielefeld, Germany. New York, NY, USA: ACM, 2014, pp. 33–40. ISBN: 978-1-4503-2658-2. DOI: 10.1145/2559636.2559677. (Visited on 10/22/2019).

[26] Wesley P Chan et al. "Design and Implementation of a Human-Robot Joint Action Framework using Augmented Reality and Eye Gaze". In: *n Proceedings of Transactions on Human-Robot Interaction* (2022).

[27] Kishan Chandan, Xiang Li, and Shiqi Zhang. "Negotiation-based Human-Robot Collaboration via Augmented Reality". In: *arXiv preprint arXiv:1909.11227* (2019).

[28] Kishan Chandan et al. "ARROCH: Augmented reality for robots collaborating with a human". In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2021, pp. 3787–3793.

[29] Chih-Wei Chang et al. "Improving the authentic learning experience by integrating robots into the mixed-reality environment". In: *Computers & Education* 55.4 (2010), pp. 1572–1578.

[30] Mark Cheli et al. "Towards an augmented reality framework for k-12 robotics education". In: *Proceedings of the 1st International Workshop on Virtual, Augmented, and Mixed Reality for HRI (VAM-HRI)*. 2018.

[31] Jonathan Wun Shiung Chong et al. "Robot programming using augmented reality: An interactive method for planning collision-free paths". In: *Robotics and Computer-Integrated Manufacturing* 25.3 (2009), pp. 689–701.

[32] Herbert H. Clark. *Arenas of language use*. University of Chicago Press, 1992.

[33] Andre Cleaver et al. "Dynamic Path Visualization for Human-Robot Collaboration". In: *Companion of the 2021 ACM/IEEE International Conference on Human-Robot Interaction*. 2021, pp. 339–343.

[34] Andre Cleaver et al. "HAVEN: A Unity-based Virtual Robot Environment to Showcase HRI-based Augmented Reality". In: *arXiv preprint arXiv:2011.03464* (2020).

[35] Andre Cleaver et al. "SENSAR: A visual tool for intelligent robots for collaborative human-robot interaction". In: *arXiv preprint arXiv:2011.04515* (2020).

[36] Robert Codd-Downey et al. “From ROS to unity: Leveraging robot and virtual environment middleware for immersive teleoperation”. In: *2014 IEEE (ICIA)*. IEEE. 2014, pp. 932–936.

[37] David Coleman et al. “Reducing the barrier to entry of complex robotic software: a moveit! case study”. In: *arXiv preprint arXiv:1404.3785* (2014).

[38] Toby HJ Collett and Bruce A MacDonald. “Augmented reality visualisation for player”. In: *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006.* IEEE. 2006, pp. 3954–3959.

[39] Francisco Cruz et al. “Explainable robotic systems: Understanding goal-driven actions in a reinforcement learning scenario”. In: *Neural Computing and Applications* (2021), pp. 1–18.

[40] Jinda Cui and Jeff Trinkle. “Toward next-generation learned robot manipulation”. In: *Science robotics* 6.54 (2021), eabd9461.

[41] Scott G Dacko. “Enabling smart retail settings via mobile augmented reality shopping apps”. In: *Technological forecasting and social change* 124 (2017), pp. 243–256.

[42] Kerstin Dautenhahn and Aude Billard. “Human–robot interaction”. In: *Wiley encyclopedia of electrical and electronics engineering* (2005).

[43] Maartje MA De Graaf et al. *Introduction to the Special Issue on Explainable Robotic Systems*. 2021.

[44] Francesco De Pace et al. “A systematic review of Augmented Reality interfaces for collaborative industrial robots”. In: *Computers & Industrial Engineering* 149 (2020), p. 106806.

[45] Jeffrey Delmerico et al. “Spatial computing and intuitive interaction: Bringing mixed reality and robotics together”. In: *IEEE Robotics & Automation Magazine* 29.1 (2022), pp. 45–57.

[46] Maximilian Diehl et al. “Augmented reality interface to verify robot learning”. In: *2020 29th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE. 2020, pp. 378–383.

[47] Griffin Dietz et al. “ARtonomous: Introducing Middle School Students to Reinforcement Learning Through Virtual Robotics”. In: *Interaction Design and Children*. 2022, pp. 430–441.

[48] Chau Do and Wolfram Burgard. “Accurate pouring with an autonomous robot using an rgb-d camera”. In: *Intelligent Autonomous Systems 15: Proceedings of the 15th International Conference IAS-15*. Springer. 2019, pp. 210–221.

[49] Zorica A Dodevska, Vladimir Kvrgić, and Velimir Štavljanin. “Augmented Reality and Internet of Things–Implementation in Projects by Using Simplified Robotic Models”. In: *European Project Management Journal* 8.2 (2018), pp. 27–35.

[50] Anca D Dragan, Kenton CT Lee, and Siddhartha S Srinivasa. “Legibility and predictability of robot motion”. In: *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE. 2013, pp. 301–308.

[51] Anca D. Dragan, Kenton C.T. Lee, and Siddhartha S. Srinivasa. “Legibility and predictability of robot motion”. In: *2013 8th ACM/IEEE Int. Con. on Human-Robot Interaction (HRI)*. ISSN: 2167-2148. Mar. 2013, pp. 301–308. DOI: 10.1109/HRI.2013.6483603.

[52] Jonatan S Dyrstad et al. “Teaching a robot to grasp real fish by imitation learning from a human supervisor in virtual reality”. In: *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2018, pp. 7185–7192.

[53] HC Fang, Soh-Khim Ong, and Andrew YC Nee. “Novel AR-based interface for human-robot interaction and visualization”. In: *Advances in Manufacturing* 2.4 (2014), pp. 275–288.

[54] Rolando Fernandez et al. “Passive Demonstrations of Light-Based Robot Signals for Improved Human Interpretability”. In: *2018 27th IEEE Int. Symp.(RO-MAN)*. IEEE. 2018, pp. 234–239.

[55] Paolo Fiorini and Francesca Renda. “Augmented reality in neurosurgery: A systematic review”. In: *Neurosurgical Review* 40.4 (2017), pp. 537–548.

[56] Terrence Fong, Illah Nourbakhsh, and Kerstin Dautenhahn. “A survey of socially interactive robots”. In: *Robotics and autonomous systems* 42.3-4 (2003), pp. 143–166.

[57] Jodi Forlizzi, Carl DiSalvo, and Francine Gemperle. “Assistive Robotics and an Ecology of Elders Living Independently in Their Homes”. In: *Hum.-Comput. Interact.* 19.1 (June 2004), pp. 25–59. ISSN: 0737-0024. DOI: 10.1207/s15327051hci1901&2_3. URL: https://doi.org/10.1207/s15327051hci1901&2_3 (visited on 11/21/2019).

[58] Jared A Frank, Matthew Moorhead, and Vikram Kapila. “Mobile Mixed-reality interfaces That enhance human–robot interaction in shared spaces”. In: *Frontiers in Robotics and AI* 4 (2017), p. 20.

[59] Joakim Grant Frederiksen et al. “Cognitive load and performance in immersive virtual reality versus conventional virtual reality simulation training of laparoscopic surgery: a randomized trial”. In: *Surgical endoscopy* 34 (2020), pp. 1244–1252.

[60] Amos Freedy et al. “Measurement of trust in human-robot collaboration”. In: *2007 International symposium on collaborative technologies and systems*. Ieee. 2007, pp. 106–114.

[61] Chiara Gabellieri et al. “Grasp it like a pro: Grasp of unknown objects with robotic hands based on skilled human expertise”. In: *IEEE Robotics and Automation Letters* 5.2 (2020), pp. 2808–2815.

[62] Samir Yitzhak Gadre et al. “End-User Robot Programming Using Mixed Reality”. en. In: *2019 (ICRA)*. Montreal, QC, Canada: IEEE, May 2019,

pp. 2707–2713. ISBN: 978-1-5386-6027-0. DOI: 10.1109/ICRA.2019.8793988. (Visited on 10/14/2019).

[63] Samir Yitzhak Gadre et al. "End-user robot programming using mixed reality". In: *2019 International conference on robotics and automation (ICRA)*. IEEE. 2019, pp. 2707–2713.

[64] Abraham Prieto García et al. "Mixed reality educational environment for robotics". In: *2011 IEEE International Conference on Virtual Environments, Human-Computer Interfaces and Measurement Systems Proceedings*. IEEE. 2011, pp. 1–6.

[65] György Gergely et al. "Taking the intentional stance at 12 months of age". In: *Cognition* 56.2 (1995), pp. 165–193.

[66] James J Gibson. "The ecological approach to the visual perception of pictures". In: *Leonardo* 11.3 (1978), pp. 227–235.

[67] Michael J Gielniak and Andrea L Thomaz. "Generating anticipation in robot motion". In: *2011 RO-MAN*. IEEE. 2011, pp. 449–454.

[68] Jennifer Goetz and Sara Kiesler. "Matching robot appearance and behavior to tasks to improve human-robot cooperation". In: *Proceedings of the 12th IEEE international workshop on Robot and human interactive communication* (2003), pp. 55–60.

[69] Michael A Goodrich and Alan C Schultz. "Human-robot interaction: a survey". In: *Foundations and Trends® in Human-Computer Interaction* 1.3 (2007), pp. 203–275.

[70] Maartje MA de Graaf et al. "Explainable robotic systems". In: *Companion of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*. 2018, pp. 387–388.

[71] Scott A Green et al. "Human-robot collaboration: A literature review and augmented reality approach in design". In: *International journal of advanced robotic systems* 5.1 (2008), p. 1.

[72] Scott A. Green et al. "Human-Robot Collaboration: A Literature Review and Augmented Reality Approach in Design". en. In: *International Journal of Advanced Robotic Systems* 5.1 (Mar. 2008), p. 1. ISSN: 1729-8814, 1729-8814. DOI: 10.5772/5664. (Visited on 09/30/2019).

[73] Thomas R Groechel et al. "Reimagining rviz: Multidimensional augmented reality robot signal design". In: *2022 31st IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE. 2022, pp. 1224–1231.

[74] Uwe Gruenefeld et al. "Mind the ARm: Realtime visualization of robot motion intent in head-mounted augmented reality". In: *Proceedings of the Conference on Mensch und Computer*. 2020, pp. 259–266.

[75] Tatiana López Guevara et al. "Adaptable pouring: Teaching robots not to spill using fast but approximate fluid simulation". In: *1st Conference on Robot Learning 2017*. 2017, pp. 77–86.

[76] David Gunning et al. "XAI—Explainable artificial intelligence". In: *Science robotics* 4.37 (2019), eaay7120.

[77] Azra Habibovic et al. "Communicating intent of automated vehicles to pedestrians". In: *Frontiers in psychology* 9 (2018), p. 1336.

[78] Jessica Hamrick, Peter Battaglia, and Joshua B Tenenbaum. "Internal physics models guide probabilistic judgments about object dynamics". In: *Proceedings of the 33rd annual conference of the cognitive science society*. Vol. 2. Cognitive Science Society Austin, TX. 2011.

[79] Lane Harrison et al. "Non-verbal communication in autonomous virtual agents". In: *Autonomous Agents and Multi-Agent Systems* 24.1 (2012), pp. 1–36.

[80] Sandra G Hart and Lowell E Staveland. "Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research". In: *Advances in psychology*. Vol. 52. Elsevier, 1988, pp. 139–183.

[81] Sunao Hashimoto et al. “Touchme: An augmented reality based remote robot manipulation”. In: *The 21st International Conference on Artificial Reality and Telexistence, Proceedings of ICAT2011*. Vol. 2. 2011.

[82] Hooman Hedayati, Mark D Gross, and Daniel Szafir. “What Information Should a Robot Convey?” In: *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2021, pp. 6232–6239.

[83] Jeffrey Heer and Ben Shneiderman. “Interactive dynamics for visual analysis: A taxonomy of tools that support the fluent and flexible use of visualizations”. In: *Queue* 10.2 (2012), pp. 30–55.

[84] Germund Hesslow. “Conscious thought as simulation of behaviour and perception”. In: *Trends in cognitive sciences* 6.6 (2002), pp. 242–247.

[85] Nathan G Hockstein et al. “Snobot: A mobile robotic sentinel for monitoring wildlife”. In: *Journal of Field Robotics* 24.4 (2007), pp. 229–244.

[86] Kai-Feng Hsiao et al. “Applying augmented reality to enhance learning: A study of different teaching materials”. In: *Educational Technology & Society* 15.3 (2012), pp. 3–13.

[87] Yongqiang Huang, Juan Wilches, and Yu Sun. “Robot gaining accurate pouring skills through self-supervised learning and generalization”. In: *Robotics and Autonomous Systems* 136 (2021), p. 103692.

[88] Bryce Ikeda. “AR Indicators for Visually Debugging Robots”. In: *2022 17th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE. 2022, pp. 1161–1163.

[89] Bryce Ikeda and Daniel Szafir. “Advancing the design of visual debugging tools for roboticists”. In: *2022 17th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE. 2022, pp. 195–204.

[90] Arthur Juliani et al. “Unity: A general platform for intelligent agents”. In: *arXiv preprint arXiv:1809.02627* (2018).

[91] Marcel Adam Just and Sashank Varma. "The organization of thinking: What functional brain imaging reveals about the neuroarchitecture of complex cognition". In: *Cognitive, Affective, & Behavioral Neuroscience* 7.3 (2007), pp. 153–191.

[92] Peter H Kahn Jr et al. "Robotic pets in the lives of preschool children". In: *Interaction Studies* 9.2 (2008), pp. 200–230.

[93] Hyeong Ryeol Kam et al. "Rviz: a toolkit for real domain data visualization". In: *Telecommunication Systems* 60 (2015), pp. 337–345.

[94] Monroe Kennedy et al. "Autonomous precision pouring from unknown containers". In: *IEEE Robotics and Automation Letters* 4.3 (2019), pp. 2317–2324.

[95] Monroe Kennedy et al. "Precise dispensing of liquids using visual feedback". In: *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE. 2017, pp. 1260–1266.

[96] Oussama Khatib et al. "Human–robot interaction". In: *The International Journal of Robotics Research* 18.7 (1999), pp. 693–713.

[97] S. Kiesler. "Fostering common ground in (HRI)". In: *ROMAN 2005. IEEE Int. Workshop on Robot and Human Interactive Communication, 2005.* ISSN: 1944-9445, 1944-9437. Aug. 2005, pp. 729–734. DOI: 10.1109/ROMAN.2005.1513866.

[98] Günther Knoblich, Stephen Butterfill, and Natalie Sebanz. "Psychological research on joint action: theory and data". In: *Psychology of learning and motivation* 54 (2011), pp. 59–101.

[99] Konrad P Körding and Daniel M Wolpert. "Bayesian integration in sensorimotor learning". In: *Nature* 427.6971 (2004), pp. 244–247.

[100] James Kubricht et al. "Probabilistic Simulation Predicts Human Performance on Viscous Fluid-Pouring Problem." In: *CogSci*. 2016.

[101] Sonya S Kwak. "The impact of the robot appearance types on social interaction with a robot and service evaluation of a robot". In: *Archives of Design Research* 27.2 (2014), pp. 81–93.

[102] Kangwoo Lee et al. "Bring common ground into robotics". In: *Int. Journal of Humanoid Robotics* 08.03 (Sept. 2011), pp. 607–629. ISSN: 0219-8436. DOI: 10.1142/S021984361100254X. URL: https://www.worldscientific.com/doi/abs/10.1142/S021984361100254X (visited on 10/23/2019).

[103] Ian Lenz, Honglak Lee, and Ashutosh Saxena. "Deep learning for detecting robotic grasps". en. In: *The Int. Journal of Robotics Research* 34.4-5 (Apr. 2015). Publisher: SAGE Publications Ltd STM, pp. 705–724. ISSN: 0278-3649. DOI: 10.1177/0278364914549607. (Visited on 02/27/2020).

[104] Matthew B Luebbers et al. "Arc-lfd: Using augmented reality for interactive long-term robot skill maintenance via constrained learning from demonstration". In: *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2021, pp. 3794–3800.

[105] Matthew B Luebbers et al. "Augmented reality interface for constrained learning from demonstration". In: *Proceedings of the 2nd International Workshop on Virtual, Augmented and Mixed Reality for HRI (VAM-HRI)*. 2019.

[106] Zhanat Makhataeva and Huseyin Atakan Varol. "Augmented reality for robotics: A review". In: *Robotics* 9.2 (2020), p. 21.

[107] Guido Makransky, Thomas S Terkildsen, and Richard E Mayer. "Adding immersive virtual reality to a science lab simulation causes more presence but less learning". In: *Learning and instruction* 60 (2019), pp. 225–236.

[108] Steve Mann. "Wearable computing: A first step toward personal imaging". In: *Computer* 30.2 (1997), pp. 25–32.

[109] William A Mattingly et al. "Robot design using Unity for computer games and robotic simulations". In: *2012 17th (CGAMES)*. IEEE. 2012, pp. 56–59.

[110] George Michalos et al. “Augmented reality (AR) applications for supporting human-robot interactive cooperation”. In: *Procedia CIRP* 41 (2016), pp. 370–375.

[111] P. Milgram et al. “Applications of augmented reality for human-robot communication”. In: *Proceedings of 1993 IEEE/RSJ Int. Con. on Intelligent Robots and Sys. (IROS '93)*. Vol. 3. ISSN: null. July 1993, 1467–1472 vol.3. DOI: 10.1109/IROS.1993.583833.

[112] Yoshiaki Mizuchi and Tetsunari Inamura. “Cloud-based multimodal human-robot interaction simulator utilizing ROS and unity frameworks”. In: *2017 IEEE/SICE (SII)*. IEEE. 2017, pp. 948–955.

[113] T. Moore and D. Stouch. “A Generalized Extended Kalman Filter Implementation for the Robot Operating System”. In: *Proceedings of the 13th Int. Con. on Intelligent Autonomous Sys. (IAS-13)*. Springer, July 2014.

[114] Christopher T Morley et al. “Mixed Reality Surgical Navigation System; Positional Accuracy Based on Food and Drug Administration Standard”. In: *Surgical Innovation* 31.1 (2024), pp. 48–57.

[115] D. Mourtzis, V. Zogopoulos, and E. Vlachou. “Augmented Reality Application to Support Remote Maintenance as a Service in the Robotics Industry”. In: *Procedia CIRP*. Manufacturing Systems 4.0 – Proceedings of the 50th CIRP Conference on Manufacturing Systems 63 (Jan. 2017), pp. 46–51. ISSN: 2212-8271. DOI: 10.1016/j.procir.2017.03.154. (Visited on 10/12/2019).

[116] Damian Mrowca et al. “Flexible neural representation for physics prediction”. In: *Advances in neural information processing systems* 31 (2018).

[117] Faizan Muhammad et al. “Creating a Shared Reality with Robots”. In: *2019 14th ACM/IEEE Int. Con. on (HRI)*. ISSN: 2167-2121. Mar. 2019, pp. 614–615. DOI: 10.1109/HRI.2019.8673191.

[118] Faizan Muhammad et al. “Creating a shared reality with robots”. In: *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE. 2019, pp. 614–615.

[119] Bilge Mutlu et al. “Perceptions of robot gestures and proxemics”. In: *Proceedings of the 4th ACM/IEEE international conference on Human robot interaction*. 2009, pp. 61–68.

[120] Vivek Myers et al. “Learning multimodal rewards from rankings”. In: *Conference on Robot Learning*. PMLR. 2022, pp. 342–352.

[121] Rhys Newbury et al. “Visualizing robot intent for object handovers with augmented reality”. In: *arXiv preprint arXiv:2103.04055* (2021).

[122] Stefanos Nikolaidis et al. “Planning with verbal communication for human-robot collaboration”. In: *ACM Transactions on Human-Robot Interaction (THRI)* 7.3 (2018), pp. 1–21.

[123] Yusuke Okuno et al. “Providing route directions: design of robot’s utterance, gesture, and timing”. In: *2009 4th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE. 2009, pp. 53–60.

[124] Soh-Khim Ong et al. “Augmented reality-assisted robot programming system for industrial applications”. In: *Robotics and Computer-Integrated Manufacturing* 61 (2020), p. 101820.

[125] Valerio Ortenzi et al. “Object handovers: a review for robotics”. In: *IEEE Transactions on Robotics* 37.6 (2021), pp. 1855–1873.

[126] Elena Pacchierotti, Henrik I Christensen, and Patric Jensfelt. “Human-robot embodied interaction in hallway settings: a pilot user study”. In: *ROMAN 2005. IEEE Int. Workshop on Robot and Human Interactive Communication, 2005*. IEEE. 2005, pp. 164–171.

[127] Tim Paek and Eric Horvitz. “Uncertainty, Utility, and Misunderstanding: A Decision-Theoretic Perspective on Grounding in Conversational Sys.” In: 1999.

[128] Andreas Paepcke and Leila Takayama. "Exploring the effects of human-like motion on observational learning in robots". In: *Proceedings of the 5th ACM/IEEE international conference on Human-robot interaction* (2010), pp. 61–68.

[129] Riccardo Palmarini et al. "Designing an AR interface to improve trust in Human-Robots collaboration". In: *Procedia CIRP* 70 (2018), pp. 350–355.

[130] Zherong Pan, Chonhyon Park, and Dinesh Manocha. "Robot motion planning for pouring liquids". In: *Proceedings of the international conference on automated planning and scheduling*. Vol. 26. 2016, pp. 518–526.

[131] Christina Pasalidou and Nikolaos Fachantidis. "Contextualizing Educational Robotics Programming with Augmented Reality". In: *2022 8th International Conference of the Immersive Learning Research Network (iLRN)*. IEEE. 2022, pp. 1–5.

[132] Max Pascher et al. "How to Communicate Robot Motion Intent: A Scoping Review". In: *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 2023, pp. 1–17.

[133] Stephen D Peterson et al. "Visual clutter management in augmented reality: Effects of three label separation methods on spatial judgments". In: *2009 IEEE Symposium on 3D User Interfaces*. IEEE. 2009, pp. 111–118.

[134] David Premack and Guy Woodruff. "Does the chimpanzee have a theory of mind?" In: *Behavioral and Brain Sciences* 1.4 (1978), pp. 515–526.

[135] Rodrigo Chacón Quesada and Yiannis Demiris. "Holo-SpoK: Affordance-Aware Augmented Reality Control of Legged Manipulators". In: *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2022, pp. 856–862.

[136] Morgan Quigley et al. "ROS: an open-source Robot Operating System". In: *ICRA workshop on open source software*. Vol. 3. 3.2. Kobe, Japan. 2009, p. 5.

[137] Camilo Perez Quintero et al. "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.

[138] Ana C Huamán Quispe, Eric Martinson, and Kentaro Oguchi. "Learning user preferences for robot-human handovers". In: *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2017, pp. 834–839.

[139] Iulian Radu, Vivek Hv, and Bertrand Schneider. "Unequal impacts of augmented reality on learning and collaboration during robot programming with peers". In: *Proceedings of the ACM on Human-Computer Interaction* 4.CSCW3 (2021), pp. 1–23.

[140] Aswin K Ramasubramanian et al. "Digital Twin for Human–Robot Collaboration in Manufacturing: Review and Outlook". In: *Applied Sciences* 12.10 (2022), p. 4811.

[141] Deepak Rao et al. "Grasping novel objects with depth segmentation". In: *2010 IEEE/RSJ Int. Con. on Intelligent Robots and Sys.* ISSN: 2153-0858. Oct. 2010, pp. 2578–2585. DOI: 10.1109/IROS.2010.5650493.

[142] Eric Rosen et al. "Bridging the Semantic Gap for Robots: Action-Oriented Semantic Maps via Mixed Reality". In: ().

[143] Eric Rosen et al. "Communicating and controlling robot arm motion intent through mixed-reality head-mounted displays". en. In: *The Int. Journal of Robotics Research* 38.12-13 (Oct. 2019), pp. 1513–1526. ISSN: 0278-3649. DOI: 10.1177/0278364919842925. (Visited on 10/11/2019).

[144] Eric Rosen et al. "Communicating and controlling robot arm motion intent through mixed-reality head-mounted displays". In: *The International Journal of Robotics Research* 38.12-13 (2019), pp. 1513–1526.

[145] Leonel Rozo, Pablo Jiménez, and Carme Torras. "Force-based robot learning of pouring skills using parametric hidden markov models". In: *9th International Workshop on Robot Motion and Control*. IEEE. 2013, pp. 227–232.

[146] Jonas Schmidtler et al. "A questionnaire for the evaluation of physical assistive devices (quead): Testing usability and acceptance in physical human-robot interaction". In: *2017 ieee international conference on systems, man, and cybernetics (smc)*. IEEE. 2017, pp. 876–881.

[147] Rossitza Setchi, Maryam Banitalebi Dehkordi, and Juwairiya Siraj Khan. "Explainable robotics in human-robot interactions". In: *Procedia Computer Science* 176 (2020), pp. 3057–3066.

[148] Megha Sharma et al. "Communicating affect via flight path Exploring use of the Laban Effort System for designing affective locomotion paths". In: *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE. 2013, pp. 293–300.

[149] Thomas B Sheridan. "Human–robot interaction: status and challenges". In: *Human factors* 58.4 (2016), pp. 525–532.

[150] Moondeep C Shrestha et al. "Communicating directional intent in robot navigation using projection indicators". In: *2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE. 2018, pp. 746–751.

[151] Emrah Akin Sisbot, Luis F Marin, and Rachid Alami. "Spatial reasoning for human robot interaction". In: *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE. 2007, pp. 2281–2287.

[152] Alexander Skulmowski. "Guidelines for choosing cognitive load measures in perceptually rich environments". In: *Mind, Brain, and Education* 17.1 (2023), pp. 20–28.

[153] Mel Slater and Maria V Sanchez-Vives. "Enhancing our lives with immersive virtual reality". In: *Frontiers in Robotics and AI* 3 (2016), p. 74.

[154] Elizabeth S Spelke et al. “Origins of knowledge.” In: *Psychological review* 99.4 (1992), p. 605.

[155] Aaron Steinfeld et al. “Common metrics for human-robot interaction”. In: *Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction*. 2006, pp. 33–40.

[156] Mike Stilman et al. “Augmented reality for robot development and experimentation”. In: *Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, Tech. Rep. CMU-RI-TR-05-55* 2.3 (2005).

[157] Kristina Stojmenova and Jaka Sodnik. “Detection-response task—uses and limitations”. In: *Sensors* 18.2 (2018), p. 594.

[158] Francesca Stramandinoli et al. “Robot learning from human demonstration in virtual reality”. In: *Proceedings of the 1st International Workshop on Virtual, Augmented, and Mixed Reality for HRI (VAM-HRI)*. 2018.

[159] Ivan E Sutherland et al. “The ultimate display”. In: *Proceedings of the IFIP Congress*. Vol. 2. 506-508. New York. 1965, pp. 506–508.

[160] Ryo Suzuki et al. “Augmented reality and robotics: A survey and taxonomy for ar-enhanced human-robot interaction and robotic interfaces”. In: *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. 2022, pp. 1–33.

[161] Daniel Szafir, Bilge Mutlu, and Terrence Fong. “Communication of intent in assistive free flyers”. In: *Proceedings of the 2014 ACM/IEEE international conference on Human-robot interaction*. 2014, pp. 358–365.

[162] Minija Tamosiunaite et al. “Learning to pour with a robot arm combining goal and shape learning for dynamic movement primitives”. In: *Robotics and Autonomous Systems* 59.11 (2011), pp. 910–922.

[163] Markus Tatzgern et al. “Adaptive information density for augmented reality displays”. In: *2016 IEEE Virtual Reality (VR)*. IEEE. 2016, pp. 83–92.

[164] Markus Tatzgern et al. “Adaptive information density for augmented reality displays”. In: *2016 IEEE Virtual Reality (VR)*. ISSN: 2375-5334. Mar. 2016, pp. 83–92. DOI: 10.1109/VR.2016.7504691.

[165] Ada V Taylor et al. “Diminished reality for close quarters robotic telemanipulation”. In: *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2020, pp. 11531–11538.

[166] Sam Thellman and Tom Ziemke. “The perceptual belief problem: Why explainability is a tough challenge in social robotics”. In: *ACM Transactions on Human-Robot Interaction (THRI)* 10.3 (2021), pp. 1–15.

[167] Andrea Thomaz, Guy Hoffman, and Maya Cakmak. “Computational Human-Robot Interaction”. en. In: *Foundations and Trends in Robotics* 4.2-3 (2016), pp. 104–223. ISSN: 1935-8253, 1935-8261. DOI: 10.1561/2300000049. URL: http://www.nowpublishers.com/article/Details/ROB-049 (visited on 09/18/2019).

[168] Mathias Ciarlo Thorstensen. “Visualization of robotic sensor data with augmented reality”. MA thesis. 2017.

[169] Robert B Trelease. “Augmented reality: A class of displays on the reality-virtuality continuum”. In: *Proceedings of the IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. 2002, pp. 282–283.

[170] Tomer D Ullman et al. “Mind games: Game engines as an architecture for intuitive physics”. In: *Trends in cognitive sciences* 21.9 (2017), pp. 649–665.

[171] Ana M Villanueva et al. “Robotar: An augmented reality compatible teleconsulting robotics toolkit for augmented makerspace experiences”. In: *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 2021, pp. 1–13.

[172] Michael Walker et al. “Communicating Robot Motion Intent with Augmented Reality”. In: Feb. 2018, pp. 316–324. DOI: 10.1145/3171221.3171253.

[173] Michael Walker et al. "Communicating robot motion intent with augmented reality". In: *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*. 2018, pp. 316–324.

[174] Michael Walker et al. "Virtual, augmented, and mixed reality for human-robot interaction: A survey and virtual design element taxonomy". In: *ACM Transactions on Human-Robot Interaction* 12.4 (2023), pp. 1–39.

[175] Chao Wang and Anna Belardinelli. "Investigating explainable human-robot interaction with augmented reality". In: *5th International Workshop on Virtual, Augmented, and Mixed Reality for HRI*. 2022. URL: https://openreview.net/forum?id=S2zeAWt4hk5.

[176] Zhihua Wang et al. "3d-physnet: Learning the intuitive physics of non-rigid object deformations". In: *arXiv preprint arXiv:1805.00328* (2018).

[177] Colin Ware. *Information visualization: perception for design*. Morgan Kaufmann, 2019.

[178] Atsushi Watanabe et al. "Communicating robotic navigational intentions". In: *2015 IEEE/RSJ (IROS)*. IEEE. 2015, pp. 5763–5769.

[179] Daniel Weber, Enkelejda Kasneci, and Andreas Zell. "Exploiting Augmented Reality for Extrinsic Robot Calibration and Eye-based Human-Robot Collaboration." In: *HRI*. 2022, pp. 284–293.

[180] Henry M. Wellman, David Cross, and Jessica Watson. "Meta-analysis of theory-of-mind development: The truth about false belief". In: *Child development* 72.3 (2001), pp. 655–684.

[181] Tom Williams et al. "Augmented, Mixed, and Virtual Reality Enabling of Robot Deixis". en. In: *Virtual, Augmented and Mixed Reality: Interaction, Navigation, Visualization, Embodiment, and Simulation*. Ed. by Jessie Y.C. Chen and Gino Fragomeni. Lecture Notes in Computer Science. Cham: Springer Int. Publishing, 2018, pp. 257–275. ISBN: 978-3-319-91581-4. DOI: 10.1007/978-3-319-91581-4_19.

[182] Tom Williams et al. “Going beyond literal command-based instructions: Extending robotic natural language interaction capabilities”. In: *Twenty-Ninth AAAI Con. on Artificial Intelligence*. 2015.

[183] Tom Williams et al. “Mixed reality deictic gesture for multi-modal robot communication”. In: *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE. 2019, pp. 191–201.

[184] Tom Williams et al. “Virtual, augmented, and mixed reality for human-robot interaction (vam-hri)”. In: *2019 14th ACM/IEEE Int. Conf. on (HRI)*. IEEE. 2019, pp. 671–672.

[185] Siyuan Xiang, Ruoyu Wang, and Chen Feng. “Mobile projective augmented reality for collaborative robots in construction”. In: *Automation in Construction* 127 (2021), p. 103704.

[186] Akihiko Yamaguchi et al. “Learning pouring skills from demonstration and practice”. In: *2014 IEEE-RAS International Conference on Humanoid Robots*. IEEE. 2014, pp. 908–915.

[187] Yue Yin et al. “A state-of-the-art survey on Augmented Reality-assisted Digital Twin for futuristic human-centric industry transformation”. In: *Robotics and Computer-Integrated Manufacturing* 81 (2023), p. 102515.

[188] Farakh Zaman et al. “Investigating a virtual reality based subterranean scenario examining augmented reality implications for military operators”. In: *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. Vol. 65. 1. SAGE Publications Sage CA: Los Angeles, CA. 2021, pp. 1129–1133.

[189] Antonio Zea and Uwe D Hanebeck. “iviz: A ROS visualization app for mobile devices”. In: *Software Impacts* 8 (2021), p. 100057.

[190] Tianhao Zhang et al. “Deep imitation learning for complex manipulation tasks from virtual reality teleoperation”. In: *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2018, pp. 5628–5635.

[191] Ziyi Zhang et al. “An Augmented Reality Platform for Introducing Reinforcement Learning to K-12 Students with Robots”. In: *arXiv preprint arXiv:2110.04697* (2021).