RoboVisAR: Immersive Authoring of Condition-based AR Robot Visualisations

Rasmus Skovhus Lunding
Aarhus University
Denmark

Mille Skovhus Lunding
Aarhus University
Denmark

Tiare Feuchtner
University of Konstanz, Germany
Aarhus University, Denmark

Marianne Graves Petersen
Aarhus University
Denmark

Kaj Grønbæk
Aarhus University
Denmark

Ryo Suzuki
University of Calgary
Canada

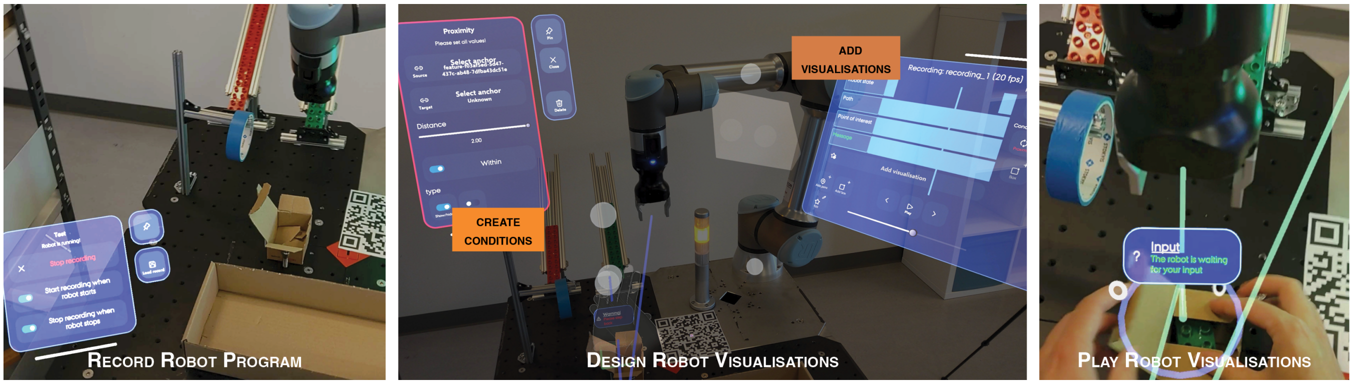
Record Robot Program
Design Robot Visualisations
Play Robot Visualisations
CREATE CONDITIONS
ADD VISUALISATIONS
Proximity
Recording: recording_1 (20 fps)
Input
The robot is waiting for your input

Figure 1: RoboVisAR allows users to author robot intent visualisations in AR. The user first records an example robot program (left), and then designs custom robot visualisations with conditions for when they appear (middle). The created visualisations can then be run in the example or in a new robot program (right).

ABSTRACT

We introduce RoboVisAR, an immersive augmented reality (AR) authoring tool for in-situ robot visualisations. AR robot visualisations, such as the robot's movement path, status, and safety zones, have been shown to benefit human-robot collaboration. However, their creation requires extensive skills in both robotics and AR programming. To address this, RoboVisAR allows users to create custom AR robot visualisations without programming. By recording an example robot behaviour, users can design, combine, and test visualisations in-situ within a mixed reality environment. RoboVisAR currently supports six types of visualisations (Path, Point of Interest, Safety Zone, Robot State, Message, Force/Torque) and four types of conditions for when they are displayed (Robot State, Proximity, Box, Force/Torque). With this tool, users can easily present different visualisations on demand and make them context-aware to avoid visual clutter. An expert user study with three participants suggests that users appreciate the customizability of the visualisations, which could easily be authored in less than ten minutes.

CCS CONCEPTS

- **Human-centered computing** → **Mfixed / augmented reality**;
- **Computer systems organization** → Robotics.

KEYWORDS

Human-Robot Collaboration; Human-Robot Interaction; Augmented Reality; Authoring

ACM Reference Format:
Rasmus Skovhus Lunding, Mille Skovhus Lunding, Tiare Feuchtner, Marianne Graves Petersen, Kaj Grønbæk, and Ryo Suzuki. 2024. RoboVisAR: Immersive Authoring of Condition-based AR Robot Visualisations. In *Proceedings of the 2024 ACM/IEEE International Conference on Human-Robot Interaction (HRI '24), March 11–14, 2024, Boulder, CO, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3610977.3634972

1 INTRODUCTION

Robots are increasingly used to automate tasks in the manufacturing industry. In particular, collaborative robots (cobots)[29] are gaining popularity in many application scenarios, as they allow human workers to work alongside, without the need for fences and other safety barriers [13]. However, studies find that cobots are not being utilised to their full potential, as they are often treated merely as uncaged industrial robots [38, 39]. In order to harness the full potential of cobots and truly achieve the human-robot collaboration (HRC), it is essential to support better communication and convey an understanding of the robot's real-time intentions [38].

HRI '24, March 11–14, 2024, Boulder, CO, USA

ACM ISBN 979-8-4007-0322-5/24/03.
https://doi.org/10.1145/3610977.3634972

In fact, understanding the robot's intentions has gained increasing attention within human-robot interaction (HRI) [44]. Researchers have explored various ways to communicate the robot's intent such as using lights [49], anthropomorphic effects [18, 21], and haptic feedback [42]. Recently, augmented reality (AR) has in particular been explored as a useful technology to communicate the robot's intent and visualise safety information [15, 16, 23, 26, 47]. With these systems, visualising the robot's intent has been shown to increase user awareness [20], the feeling of safety [20, 37], and the user's understanding of the robot behaviour when working close together [36]. However, creating these AR robot visualisations requires a high level of software and robotics skills.

In this work, we present RoboVisAR, an immersive AR authoring tool to create in-situ robot visualisations. RoboVisAR allows users to create AR robot visualisations within a mixed reality environment (HoloLens 2) without the need for programming. To author visualisations, users first record an example of the robot's behaviour, which then serves as the basis to explore different AR visualisations. Users can rapidly scroll through the recorded data to see the robot movement on a virtual robot replica. To this the user can add different visualisations and instantly see how they appear throughout the robot program. Once satisfied with the authored visualisations, users can test these AR visualisations by applying them to the physical robot in their workspace. To enable the easy and quick authoring process, RoboVisAR supports six types of common AR robot *visualisations*: Path, Point of Interest, Robot State, Message, Safety Zone, and Force/Torque. Each type of visualisation has different properties to customise its visual appearance. Furthermore, the system has a custom condition feature, which allows users to define when these visualisations appear. RoboVisAR supports four types of *conditions*: Robot State, Proximity, Box, and Force/Torque. By combining these conditions and visualisation types, users can easily author various AR visualisations, such as showing the safety zone when the user approaches the robot, or rendering a movement trajectory path for a certain task.

To evaluate our system, we conducted an expert user study with three participants. Based on a predefined robot program, participants were asked to perform two tasks: 1) create a predefined set of visualisations, and 2) freely design and explore own visualisations. Results from our study show that users can create useful AR robot visualisations within 10 minutes. Furthermore, we found that the immersive authoring approach is more useful for testing visualisations within the context, compared to traditional authoring workflows that require programming environments like Unity3D.

In summary, we contribute with:

(1) RoboVisAR, a novel immersive authoring tool to create in-situ robot visualisation using real data directly from a robot.
(2) A condition-based approach to robot visualisations that allows users to define when they appear.
(3) Results from an expert user study that highlight opportunities for immersive AR robot visualisations.

2 RELATED WORK

In the following we discuss related work from three areas: (1) AR robot visualisations to support human-robot interaction (HRI), (2) authoring tools within HRI, and (3) general AR authoring tools.

2.1 Visualisations to Support Human-Robot Interaction

For effective human-robot interaction, it is essential to comprehend and predict robot's intentions, and researchers have explored various strategies to relay these to users. These methodologies include motion-based cues, such as the robot's movement trajectory [17], visual cues, like a robot's "gaze" direction [40], and auditory or light signals [7, 34]. More recently, Pascher et al. [44] conducted a review on *how to communicate robot motion intent*, where AR was found to be a prominent medium. Furthermore, they proposed four different types of robot intent that can express a robots intentions to a human co-worker: *motion*, *attention*, *state*, and *instruction. Motion intent* visualises the future motion of the robot, e,g., as a movement path, *attention intent* seeks to draw human attention before an action is initiated, *state intent* communicates internal state, such as the robot's current mode (moving or stopped) or data from sensors, and finally *instruction intent* communicates instructions for assisting the robot in continuing its task. Pascher et al. further argue that robot motion intent is a rather vague term, which has not been clearly defined yet [44].

In general, augmented reality has been shown to be a promising technology for communicating robot intent to humans. Suzuki et al. [48] outline five domains where AR can enhance human-robot interaction: facilitate programming, support real-time control, improve safety, communicate intent, and increase expressiveness [48]. For example, Andersen et al. [1] found that projection-based intent visualisations were perceived more positively by participants collaborating with a robot than traditional text- or screen-based interfaces. Rosen et al. [47] investigated AR-based visualisations to communicate robot motion intent for an obstacle avoidance task. Comparing AR visualisations on a head-mounted display (HMD), visualisations on a 2D display, and no-visualisations, they found that AR can reduce the perceived workload and enhance collision predictability. Hietanen et al. [26] explored how projection-based and HMD-based AR could support human-robot collaboration, and found that AR in general can improve the collaborative experience. Other researchers have compared different AR visualisations: Gruenefeld et al. [23] compared three visualisations on an HMD, Arevalo Arboleda et al. [3] compared no cues, basic cues, and advances cues for robot teleportation on a HMD, and Hetherington et al. [25] investigated three legibility cues for a moving robot. Tsamis et al. utilised multiple AR visualisations in their system, including safety bubble, robot's path, silhouette preview of the robot, and warning messages [51]. Similarly, Andronas et al. combined several different AR visualisations for human-robot collaboration tasks [2]. Lunding et al. found that users had different preferences when combining different visualisations in a single system. E.g., while some users found path visualisations useful, others found it redundant as the robot's next task was also visualised [36].

While these studies give insights about how different AR intent visualisations perform in certain tasks, it is worth noting that the results might not be universally applicable across different contexts [3, 23, 25]. This highlights the importance of developing authoring tools that allow users to easily design AR robot visualisations relevant to their specific contexts and scenarios. Furthermore, researchers have emphasised the need for creating more intuitive authoring tools to support non-technical users to develop and test different visualisations [39, 41, 48], which also applies to robot intent visualisations. Currently, the design and creation of these visualisations is largely limited to programmers. In this work, we aim to broaden the space for AR robot intent visualisations, by creating an authoring tool to allow non-technical users to participate in the development of these visualisations.

2.2 Authoring Tools for Human-Robot Interaction

Within human-robot interaction, authoring tools have primarily focused on supporting non-technical users in robot programming. Existing authoring tools include V.Ra [10], KineticAR [19], GhostAR [9], and Figaro [45]. However, most of these tools focus on the robot programming aspect, whereas our focus is on authoring visualisations to help users understand the robot’s intent. Similar to our goal, there are a few tools that focus on authoring robot visualisations. For example, RViz (short for “ROS visualisation”) is a 3D visualisation and debugging tool, that enables robot programmers to visualise the robot’s sensor data and internal state when using the ROS ecosystem [24]. In addition, several tools have been made that support visualisations in AR [4, 27, 28]. For creating robot intent visualisations, Unity’s Robotics Visualisation Package [50], allows users to create visualisations in Unity based on ROS-messages. However, these tools are embedded in programming environments and they are based on the ROS ecosystem. Hence, users need extensive technical skills to create and use the visualisations. Moreover, these tools do not support *immersive authoring* within a mixed reality environment, thus, users need to go back and forth between the computer screens and physical environment to test the visualisation. In contrast, RoboVisAR allows users to create in-situ robot intent visualisations within a mixed reality environment through an immersive authoring interface.

2.3 Augmented Reality Authoring Tools

In human-computer interaction, there has been an increased focus on augmented reality authoring tools, which enable users to create their own AR applications without programming. For example, Pronto [33], Rapido [32], and 360proto [43] allow designers to prototype new AR applications. The Meta-AR-App [52] and Augmented Math [14] allow teachers to create augmented teaching material for students, and Augmented Reality Scratch [46], ExposAR [35], and StoryMakAR [22] allow children to author their own AR applications. While these tools are a great inspiration for designing authoring tools, they do not work in the context of robot intent, which requires more seamless integration between the robot data and AR visualisations.

A few tools have explored AR authoring to integrate sensors and actuation in the physical world. For example, in CAPturAR [54] and ProGesAR [55], users can prototype context-aware IoT applications by recording the human’s movement or by using proxies through AR-based interfaces. In MechARSpace [56], users can author AR enhanced toys with a two-way biding between AR content and physical sensors and actuators from a plug-and-play IoT toolkit. While these tools do not focus on visualisation aspects, there are some that do: in PapARVis Designer [12] users can extend static visualisations in physical books by augmenting them with virtual content. The MIRIA toolkit [8] allows users to create AR visualisations for analysing interaction data in AR. However, these tools only work with pre-existing data. In our work, we focus on creating visualisations based on live data that directly comes from the robot, the human behaviour, and the physical space.

3 ROBOVISAR

This section introduces RoboVisAR, an immersive authoring tool designed for creating AR-based robot visualisations. RoboVisAR supports a rapid and iterative design process, allowing users to create and combine a diverse set of visualisations. Furthermore, the tool enables users to generate functional visualisations consistently across various robot programs and configurations.

The authoring process can be divided into the following three steps. (1) Record: capturing an example robot behaviour, (2) Design: creating timeline-based visualisations and modify their appearance with context-aware conditions based on the recorded example, and (3) Play: testing visualisations with the original or a different robot program. Consequently, RoboVisAR operates in one of three modes: *record mode*, *design mode*, or *play mode*. The user interface adapts based on the chosen mode, providing assistance tailored to the specific task. The data source shifts between *live*-data directly from the robot and other data sources in *record mode* and *play mode*, to *stored*-data of a recorded example in *design mode*. *Design mode* offers the most comprehensive interaction possibilities with the system.

Before users start authoring with RoboVisAR, basic information about the system must be configured. This includes markers, robot(s), and the name of the project. To switch between the different modes, the mode selection interface is used (Figure 2). Project setup is achieved by pressing the *Settings* button.

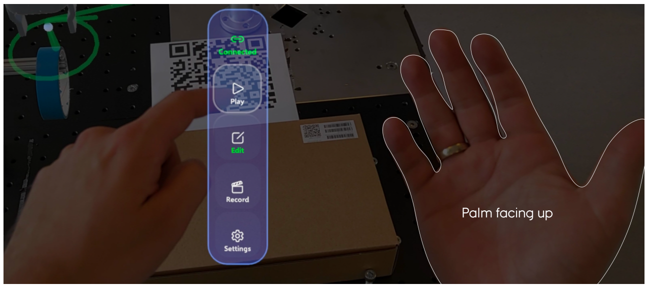

Figure 2: Screenshot of the mode selection interface. The menu is active when the palm of either hand is facing up. The menu can be interacted with using the opposite hand.

3.1 Record Mode

Before creating visualisations, RoboVisAR needs example data to initiate the authoring process. This recorded data enables previews of potential visualisations to aid the design of dynamic visualisations. Users can scroll through the recorded example to confirm the desired behaviour (Figure 4), thereby obtaining a more efficient feedback loop compared to operating the actual robot during the design process.

When *record mode* is activated, the user is asked to start a recording. While recording, the robot executes its program, and the system captures the relevant data.

Multiple recordings can be created, such as one showcasing a 'normal' execution of the robot program and another representing an 'error-case'. However, only one recording can be utilised in the *design mode* at any given moment. The recording interface is displayed in Figure 3. The record menu presented to the user when RoboVisAR is in *record mode*, which can be achieved using the mode selection interface, Figure 2.

RoboVisAR records live data directly from the robot, the HMD, and potentially other devices or software components. The following data is recorded in a generic format from the robot: robot state, joint angles, tool-centre point, force/torque values, and robot path. Furthermore, the hand and head positions of the user are recorded through the HMD. As RoboVisAR is a standalone application, all data is stored on the HMD.

Once the recording ends, either due to user intervention or automatically when the robot's state changes to *stopped*, the system transitions to the design interface, as illustrated in Figure 4.

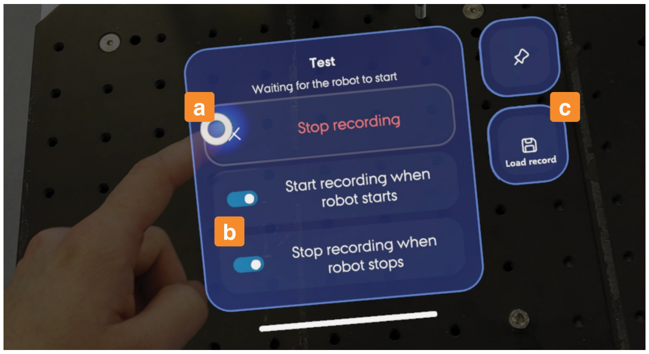
Test
Waiting for the robot to start
a
Stop recording
Start recording when robot starts
b
Stop recording when robot stops
c
Load record

Figure 3: Screenshot of the Record menu. (a) Is clicked to start/stop a recording. (b) Is an option to automatically record only when the robot is executing its program. (c) provides a list of all recordings from which the user can select. The menu is visible when in *record mode* (see Figure 2)

3.2 Design Mode

After recording, the user can transition to the *design mode* to create context-aware visualisations based on the captured behaviour. In this mode, a panel shows a timeline highlighting when specific visualisations are active within the given example, as depicted in Figure 4. Additionally, this panel offers options to add, modify, and remove visualisations, to set and adjust conditions, and to generate support objects that can serve as inputs for these conditions.

3.2.1 Exploring the Timeline. Inspired by other AR authoring tools such as Pronto [33] and Rapido [32] RoboVisAR support a timeline-view of the recorded data (see Figure 4). The timeline show which visualisations have been added and when they are active. A slider facilitates navigation to specific times within the recorded example. Additionally, the replay button allows users to preview the created visualisations on a virtual robot that reflects the robot's position during the recording.

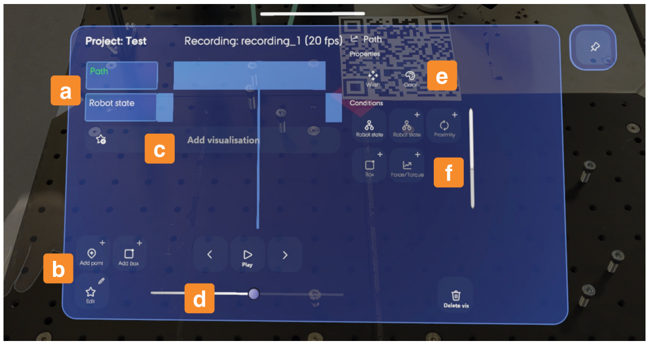

Figure 4: Screenshot of the Design panel shown in *design mode* containing: (a) a list of visualisations including the timeline showing when each is active, (b) a button to add new visualisations, (c) controls to create anchor points and new boxes, which are used for some conditions, and (d) a Play/pause button for replaying recorded data with the created visualisations. The slider can be used to navigate to specific times in the example. On the right side of the menu are (e) editable properties for the selected (Point of Interest) visualisation, and (f) conditions for the selected visualisation. Further conditions, marked with a +-symbol in the upper right corner, can be added.

3.2.2 Adding Visualisations. A user can add a new visualisations by clicking on the *add visualisation* button, as shown in Figure 4. Upon clicking, they are presented with in-situ previews of all available visualisations, based on the time frame of recorded data selected with the timeline slider. Once a selected visualisation is confirmed, it is added to the timeline and becomes available for further refinement, such as changing properties (see Figure 6) and defining conditions as described below. RoboVisAR currently supports the following six types of visualisations, which represent some of the most common approaches used in related work. An example of each can be seen in Figure 5.

- **Path visualisation**: a line indicates the robot's next movement path. Line width and colour can be modified.
- **Point of Interest**: a marker highlights a certain location of interest. This could e.g., be where the robot is about to pick or place an object. The user can edit the shape, size, line width, and line colour, and enable a pulsing animation.
- **Robot State visualisation**: a panel shows the robot's state as "Running", "Paused", or "Stopped". The user can modify the anchor and offset, i.e, where the panel is placed spatially, and whether the panel should face the user.
- **Message visualisation**: a panel that indicates either a "step-back", "help", or "wait" message. The user can again modify the anchor and offset, and whether the panel should face the user, as well as the message type.
- **Safety Zone**: a circular line indicates a safety-zone around the robot. The user can modify the line width, line colour and padding (i.e., zone size).

- **Force/Torque visualisation**: a vector indicates the current measured force or torque values. The length of the vector corresponds to the respective measured value. The user can can choose the measurement to visualise, and modify whether each axis (x, y, z) is to be shown individually or combined.

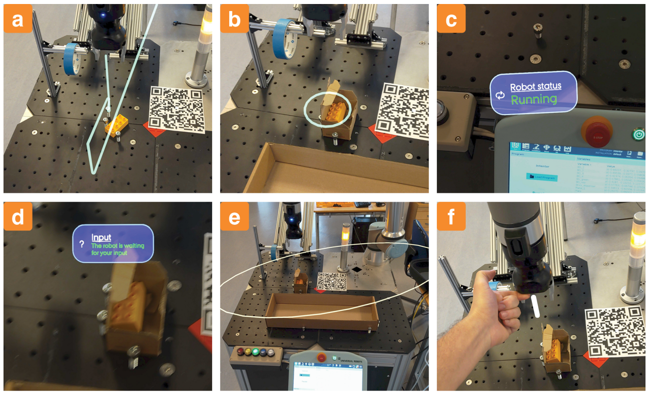

Figure 5: Screenshots of the currently implemented visualisations, showing: (a) Path, (b) Point of Interest, (c) Robot State, (d) Message, (e) Safety Zone, and (f) Force/Torque.

3.2.3 Creating Conditions. Visualisation conditions determine when a particular visualisation becomes active and visible. While all visualisations are automatically shown by default, the user can also limit the visibility of visualisations to certain times, to avoid visual clutter. To support this, RoboVisAR can currently track and trigger four types of conditions, which can be customised and combined. A visualisation appears when its corresponding condition is fulfilled based on the following:

- **Robot State Condition**: activates based on the robot's operational state ("Running", "Paused", "Stopped").
- **Proximity Condition**: becomes active when two anchor points (described below) come within a certain distance of each other.
- **Box Condition**: activates when an anchor point resides within a specified area delimited by the box. A box is a modifiable cuboid, allowing the user to define a specific volumetric area in 3D space.
- **Force/Torque Condition**: becomes active when either the measured force or torque are greater than a threshold specified for either the x, y, or z-axis or all axes combined.

All conditions have an option to *flip* when they are active. For example, the Proximity condition can be set to become active when the two anchor points are separated by a distance greater than the predefined threshold (see Figure 6, c).

Beyond controlling the visibility and activation of a visualisation, conditions can also dynamically modify properties of a visualisation when they're active. For example, the colour of a Safety Zone visualisation can change based on the robot state, to be red when running and otherwise green. Or a Point of Interest visualisation can change shape depending on the type of robot action (if the action types are physically separable), which could for example be achieved with a Box condition.

3.2.4 Adding Anchors. By default, a scene contains several anchors, which are used to establish spatial relationships. All visualisations have an anchor property and the user can redefine which anchor a visualisation is attached to. However, this is only meaningful for some visualisations. For example, a Path visualisation is always anchored to the robot's base and cannot be changed, since that would break the inherent spatial relationship between the robot and the Path visualisation. In contrast, the panel showing the Robot State, which does not have an obvious pre-existing spatial relationship to the robot, can be modified by the user.

In addition to repositioning visualisations, anchors can also be incorporated in spatial conditions, such as the Proximity and Box conditions described previously. Typical available anchors include the main environment (centred on the QR-code used to synchronise coordinate systems between the HoloLens and the robot), parts of the robot (i.e., robot base, robot tool centre point aka. "fingertips"), and body parts of the user (i.e., left hand, right hand, and head). Beyond these, users can specify new anchors. The functionality to add and modify anchors is shown in Figure 4.

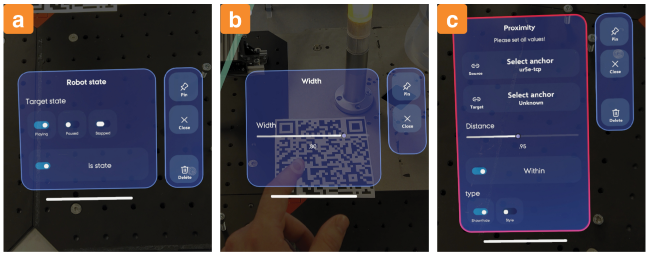

Figure 6: (a) Panel to customise the 'width' property of a visualisation, such as a Path and Point of Interest. (b) Panel to edit a Robot State condition. (c) Panel to edit a Proximity condition. The red border indicates that one or more properties need initialisation, as the second anchor is not yet set.

3.3 Play Mode

Play mode can be used to test the authored visualisations with the real robot, or simply when the authoring process is complete and the visualisations are ready for deployment. In *play mode*, data is transmitted directly from the robot to the system.

Since conditions are specified with regards to the context but not tied to a specific robot program, all visualisations should behave consistently, even if there are modifications to the robot program. Their behaviour is contingent on events (i.e., data changes) rather than on specific timelines.

If the user is unsatisfied with the outcome, they can either revert to design mode to adjust and refine existing visualisations, or create a new recording that more aptly captures the context wherein the desired behaviours should manifest.

3.4 System Implementation

Our system uses a head-mounted display (Microsoft HoloLens 2), a Universal Robot 5e (UR5e, Polyscope: v5.9.1), and a server (ASUS PN51-E1) running the MQTT-broker (mosquitto v2.0.15, MQTT v3.1.1) and UR-mqtt-adapter (v. 0.5). RoboVisAR is a Unity

(2021.3.23f1) application with the following packages: MRTK3 (pre.18), MS MixedReality QR (0.5.3037), and MQTTnet (4.2.0.706). The application runs on the HoloLens.

RoboVisAR is implemented as a framework, such that adding visualisations, conditions, and robots requires little work for the developer. For example, a condition must implement three methods: Active(Frame), IsInitialized(), and Properties(). The Active method takes a data-frame, i.e, a collection of all data at time t, as input and returns a boolean-value for whether the condition is active. IsInitialized() returns true, if all properties have meaningful values. Properties() returns a list of properties, which can be edited by the user. A new visualisation likewise requires three methods to be implemented and, optionally, to override the appropriate methods for receiving data, e.g., NewRobotState() or NewJointState().

3.4.1 Shared Coordinate System. As is well known, various systems use different conventions for describing poses. Unity3D uses a left-handed, Y-up coordinate system, whereas that in ROS is right-handed, and the coordinate system defined for the UR5e robot is right-handed and Z-up. To ensure consistent representations, we have decided that all coordinates used in RoboVisAR must be converted into the Unity3D format. Thus, all robot adaptors must handle the conversion, if necessary, before publishing messages to shared MQTT-topics.

Furthermore, to align the origins of the coordinate systems for the HMD, the robot, and the workspace, we use QR-codes. These are easy to implement on the HoloLens 2 and we have found that they afford sufficiently stable tracking, given a QR Code with sufficient level of detail.

3.4.2 Getting Data into the System (and Recording It). Since RoboVisAR depends on data captured from a real robot, it is necessary to establish a connection to the robot. This "robot adaptor" consists of a MQTT communication layer between the HMD and robot. While other options, like ROS, could offer viable alternatives, we opted for MQTT due to its simplicity and ease of implementation across platforms and programming languages.

We have defined a protocol that specifies how data should be formatted and which topic it is expected to be published to. An implementation of a robot adaptor is not required to support the entire protocol, as this may not be possible. However, this might limit the available visualisations and conditions. While we provide an adaptor for Universal Robots e-series, RoboVisAR will also support KUKA iiwa and KUKA KR120 robots, with the corresponding adaptors.

4 AR EXPERT REVIEW

To evaluate RoboVisAR (v. 0.3), we conducted an expert review study with AR professionals who have experience with HRI. Our goals were to (1) observe and assess how experts use RoboVisAR to create robot visualisations and (2) examine the qualitative usability and utility of our authoring tool. This evaluation strategy falls into the category of "usage evaluation", as described in the HCI toolkit evaluation strategy classification [31].

A more controlled experiment to compare RoboVisAR with a baseline seems difficult, as no clear baseline exist. Currently, to our knowledge and experience, robot visualisations are hand-crafted for each project and no authoring tools are available for this purpose. As RoboVisAR introduces a new concept for robot visualisations, we did not test it with novice users, but relied on experts to gain first insights. We expect that this initial feedback will help highlight the benefits and areas for improvement of RoboVisAR for future iterations.

We recruited three AR professionals (n = 3, zero female), based on their prior experience with human-robot interaction. The participants self-identified as PhD-student (P1) and software-developers (P2, P3). Their age ranged from 26 to 32 years.

4.1 Procedure

Participants were informed about the study purpose and they gave their written consent to participate. They provided background information through a questionnaire, and then received an introduction to the overall study environment and tasks, including information on the robot's built-in safety measures.

The study was split into three consecutive phases: *tutorial*, *fixed task*, and *free task*. These are described in more detail below. The study concluded with a semi-structured interview, in which participants were asked to reflect on the experience of creating the robot visualisations, to gather insights about the participants' overall conception and understanding of the interface and the task. The overall study duration was about 75 minutes on average and participants received a small compensation for their time.

4.1.1 Task 0: Tutorial. The following shared task between the robot and participant was demonstrated: the robot would pick a DUPLO brick and place it in a small cardboard box, the participant should then close and rotate the box, before pressing a button to confirm that the box was ready for the robot to pickup and place in a larger cardboard box.

After the demonstration, participants started a recording from RoboVisAR, before running the robot program once again. The tutorial then guided them through creating three different *visualisations* (Robot State, Message, and Safety Zone) in *design mode*. Each visualisation was furthermore styled and a *condition* was added to each: respectively Robot State condition, Proximity condition, and Box condition. The tutorial concluded by running the robot program once again, this time in *play mode*, to verify that the created visualisations worked as intended.

4.1.2 Task 1: Fixed task. The participants were instructed to create four visualisations (Robot State, Path, Point of Interest, and Message) based on the scenario from the tutorial. Each visualisation should have at least one property changed and one meaningful condition.

4.1.3 Task 2: Free task. A continuation of the task was demonstrated to the participant: the robot would pick and place two DUPLO bricks in the large cardboard box, while the participant was instructed to pick and place some as well. The box was then closed and flipped by the participant, before the robot simulated applying tape to seal the box. Participants spent on average 15 minutes freely exploring the opportunities of RoboVisAR to support this task.

4.2 Main findings

Our findings are based on interviews with the AR experts and can be summarised as: all participants reported that the tool works well and that they like it. In the following we report in more detail on the individual features.

The timeline was described as useful and participants especially appreciated that it shows when visualisations are active. They also valued the ability to scroll through it, e.g., to before a visualisation become active, and get a preview of how the visualisation will turn out. P2 mentioned that it resembles a video editor.

The ability to create visualisations situated directly in the application context was highlighted by all participants. Here, P2 added that it is hard to relate to the real context in e.g., Unity3D, but when using RoboVisAR, you see where the robot is and will be, where your hands are, and so on.

All participants agreed that the conditions make really good sense and they were generally happy with the available options. P2 and P3 mentioned the benefit of styling taking immediate effect, whereas their traditional workflow consists of trying out one variant at a time, with a (minutes-) long build-process in between. Further, each participant proposed some variants of new conditions that could be implemented. P1 suggested a 'robot-is-waiting-for-input'-condition. P2 suggested an 'active-when-hands-are-close'-condition, which could be achieved by automatically adding an anchor to all visualisations (new implementation) and then using the Proximity condition. P3 suggested a 'gaze'-condition.

All participants found button presses and the general menu interaction difficult with the HoloLens. Besides issues relating to mid-air interaction with HoloLens, some usability issues where reported: it is not possible to combine conditions with AND and OR (P1, P3), the colour picker is very limited (P1, P2, P3), the selection of anchors would be easier if they had a label (P1), and, finally, is it not possible to delete anchors (P3). A last improvement requested by P1 and P2 is an extension to the timeline, such that *events* from recorded data, e.g., when the robot state is changing, could be seen.

Finally, we observed that participants spend on average 9:30 minutes on *Task 1: fixed task* (P1: 8:30, P2: 7:00, P3: 13:00). It should be noted that P3 had less experience with the HoloLens and spent substantially more time struggling with button clicks for menu interaction. Overall, this suggests that with RoboVisAR users can create four visualisations, change their style, and add a condition to each visualisation within just around 10 minutes.

5 DISCUSSION

Based on our expert review study, we will discuss condition-based robot visualisations, in-situ authoring of robot visualisations, and, lastly, we will reflect on the limitations of our current work and future research directions.

5.1 Condition-based robot visualisations

In this work, we propose making the robot visualisations condition-based, such that they are only presented when needed. This allows users to combine different visualisations, while avoiding visual clutter. The idea of making content availability based on conditions has already been widely explored in other domains e.g., for proxemic interaction [5] researchers investigated how users' position and orientation can be used as input to create interactive systems. However, in the field of HRI this concept has not yet been widely applied. In most existing work on robot intent and safety visualisations (e.g., [3, 23, 26, 47]) the virtual content is perpetually visible. In other works (e.g., [2, 36, 51]) different visualisations are combined in HRI systems, and Chacko and Kapila even use sensor data to change their visualisations. We go beyond this with our condition-based approach, and allow users to easily explore when different visualisations should be presented or how they should change based on the context. Furthermore, our conditions allow the visualisations to be robust to changes in the robot program, such that they can be applied across programs or when the robot behaviour changes dynamically.

The participants in our expert study found that the implemented conditions provide a lot of opportunities. Further, they could imagine additional conditions, such as a 'gaze'-condition, or a 'robot-is-waiting-for-input'-condition to be useful. More work is needed to explore new conditions, or how users may create custom-conditions based on already available data. Another interesting extension to condition-based robot visualisations, are personalised visualisations. One could imagine that a predefined set of visualisations are created by a designer using RoboVisAR, as part of the system implementation. The system could then be extended by the operators, who modify visualisations based explicitly on their personal preferences. A slightly different approach would be a condition based on the operator's personal preferences and expertise. The relevance of this becomes apparent, when considering that, in real use, the operator's expertise will change over time.

5.2 In-situ authoring of robot visualisations

In our expert review study, participants found that creating visualisations in-situ was a great benefit, as it directly related to the context, which the visualisations was being designed for. This has also been reported by others, as it typically results in a faster feedback loop [30]. We expected the in-situ authoring to have concrete advantages, e.g., in picking suitable colours for visualisations, which are often important to make them clearly distinguishable from the environment. Another example is the verification of conditions, especially those relying on distance, which RoboVisAR facilitates.

However, the above-mentioned benefits come at some cost. For most, if not all users, interaction is still faster and more fluent with a keyboard and mouse, compared to mid-air menu interaction. An AR interface is typically also more restricted in functionality than a desktop solution would be. However, the use of gaze and gestures opens up new ways of interaction that can be more accurate and may allow more complex inputs. However, customisation to these new forms of interaction is needed, both for users but also interface developers.

The ability to easily author robot visualisations in-situ might also open new areas of exploration. RoboVisAR can make it easier to explore complementary visualisations, i.e, how different types of visualisations can effectively be used in combination. This will most likely include investigations into *redundancy*, i.e, different visualisations that convey the same information. For example, we observed in an earlier study [36] that, if a robot's task and path are both visualised separately, one of these visualisations may be perceived as redundant, since they both convey some of the same information (namely the destination of the robot). While it remains to be investigated, whether this is detrimental (e.g., due to increasing visual clutter), or whether redundancy has benefits, a promising approach may be to allow the user to toggle visualisations dynamically.

5.3 Limitations and future work

5.3.1 Limitations of the expert review study. While the recruited experts were generally excited about our system, the study had certain limitations. Firstly, the number of participants is obviously low and, consequently, the diversity of our sample is poor. This is mainly due to our requirement of the very specialised background in AR and HRI. Furthermore, the purpose of our study was to evaluate RoboVisAR, but not to investigate task performance or the overall interaction with menus and buttons on the HoloLens [36, 53, 56]. Thus, some prior experience with HoloLens was also desired. Secondly, the experts were presented with predefined robot programs in a lab-based HRI case. While this design made it possible for the experts to be exposed to most features of the system, it would be interesting to explore how the authoring tool would be used in an actual industrial HRI case. However, as participants consistently reported similar experiences, we think our study has its merits as a first evaluation of RoboVisAR, despite these limitations.

5.3.2 Hardware limitations. The current prototype is created for the HoloLens 2 HMD, which has some restrictions in both physical properties and performance. The limited field of view is often mentioned as a limitation for AR-HRI systems [6, 23]. This should be taken into consideration, when designing visualisations - at least until the current state of HMDs improves. However, extending the display area may pose new challenges, such as visual clutter, if the designers fill the available space with visualisations, without consideration of their contextual relevance. We believe RoboVisAR is a promising tool to address this issue also in future.

5.3.3 Support other robot types. The current system is designed for a particular type of robots, namely a robot arm. Although only one brand of robot has physically been tested, the system can be extended to other brands by creating respective adaptors, as described in 3.4.2. Furthermore, while many of the same visualisations are applicable for other types of robots, such as drones and mobile robots, designing specialised visualisations for these with the current system might introduce new challenges, due to the user's distance to the robots and the size of the space they are moving in. More work is needed there.

5.3.4 Supporting different modalities. In our research, we have focused on creating visual outputs. This naturally has limitations, such as information needing to be in the user's field of view to be visible, and the critical problem of visual clutter, when the user also needs to see the physical world to safely collaborate with the robot. It would be interesting to explore how visualisations can be complemented by other modalities, such as sound and haptic feedback. One could, for example, imagine that, instead of using a message to express that the robot needs the user's input, the user could be "nudged" through haptic feedback. Future research could explore how these modalities can be incorporated into the authoring process.

5.3.5 Enabling two-way communication. To foster collaboration between humans and robots, communication should not only flow from the robot to the user, but also the other way around. The already implemented conditions could play a part in such a setup: instead of controlling when a visualisation is presented to the user, they could instead trigger when a signal is sent to the robot. E.g., when the user approaches the robot, a proximity condition can trigger the robot to move at a slower speed, or the user could acknowledge a request from the robot by gazing at a virtual object.

6 CONCLUSION

In this work, we present RoboVisAR, an immersive authoring tool that enables users to create condition-based AR robot visualisations. RoboVisAR can record live data of a robot, while it is executing its program. The recorded example is then used to design condition-based AR robot visualisations in a timeline-based editor. The designed and possible additional visualisations can be tested in-situ, as they are applied to a virtual replica of the robot. Adding conditions creates an interactive flow, where visualisations are enabled or disabled based on relevant context. Through an expert review study, we showed that users can author a group of visualisations within just 10 minutes, including initial recording and playing the final result. Furthermore, participants found the in-situ authoring highly useful, described the conditions as relevant functionality, and agreed that the timeline provides a good overview of when visualisation are active. Thus, RoboVisAR provides the HRI community with a new way to author condition-based robot visualisations that are directly situated in the intended application scenario.

ACKNOWLEDGMENTS

Thanks go to Roberta, for always doing what it was told, no matter how stupid the request. This work was supported by the Innovation Fund Denmark (IFD grant no. 9090-00002B) for the MADE FAST project - part of the Manufacturing Academy of Denmark (MADE).

REFERENCES

[1] Rasmus S. Andersen, Ole Madsen, Thomas B. Moeslund, and Heni Ben Amor. 2016. Projecting robot intentions into human environments. In *2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN).* Institute of Electrical and Electronics Engineers Inc., United States, 294–301. https://doi.org/10.1109/ROMAN.2016.7745145

[2] Dionisis Andronas, George Apostolopoulos, Nikos Fourtakas, and Sotiris Makris. 2021. Multi-modal interfaces for natural Human-Robot Interaction. *Procedia Manufacturing* 54 (2021), 197–202. https://doi.org/10.1016/j.promfg.2021.07.030 10th CIRP Sponsored Conference on Digital Enterprise Technologies (DET 2020) – Digital Technologies as Enablers of Industrial Competitiveness and Sustainability.

[3] Stephanie Arevalo Arboleda, Franziska Rücker, Tim Dierks, and Jens Gerken. 2021. 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* (Yokohama, Japan) *(CHI '21).* Association for Computing Machinery, New York, NY, USA, Article 728, 14 pages. https://doi.org/10.1145/3411764.3445398

[4] Kodie Artner. 2023. AR-RViz: Augmented Reality Robot Visualization. https://github.com/kodie-artner/AR-RViz. (Accessed on 23/12/2023).

[5] Till Ballendat, Nicolai Marquardt, and Saul Greenberg. 2010. Proxemic Interaction: Designing for a Proximity and Orientation-Aware Environment. In *ACM International Conference on Interactive Tabletops and Surfaces* (Saarbrücken, Germany) *(ITS '10).* Association for Computing Machinery, New York, NY, USA, 121–130. https://doi.org/10.1145/1936652.1936676

[6] Gabriele Bolano, Yuchao Fu, Arne Roennau, and Ruediger Dillmann. 2021. Deploying Multi-Modal Communication Using Augmented Reality in a Shared Workspace. In *2021 18th International Conference on Ubiquitous Robots (UR).* IEEE, 302–307. https://doi.org/10.1109/UR52253.2021.9494689

[7] Gabriele Bolano, Arne Roennau, and Ruediger Dillmann. 2018. Transparent Robot Behavior by Adding Intuitive Visual and Acoustic Feedback to Motion Replanning. In *2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 1075–1080. https://doi.org/10.1109/ROMAN.2018.8525671

[8] Wolfgang Büschel, Anke Lehmann, and Raimund Dachselt. 2021. MIRIA: A Mixed Reality Toolkit for the In-Situ Visualization and Analysis of Spatio-Temporal Interaction Data. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 470, 15 pages. https://doi.org/10.1145/3411764.3445651

[9] Yuanzhi Cao, Tianyi Wang, Xun Qian, Pawan S. Rao, Manav Wadhawan, Ke Huo, and Karthik Ramani. 2019. GhostAR: A Time-Space Editor for Embodied Authoring of Human-Robot Collaborative Task with Augmented Reality. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology* (New Orleans, LA, USA) *(UIST '19)*. Association for Computing Machinery, New York, NY, USA, 521–534. https://doi.org/10.1145/3332165.3347902

[10] Yuanzhi Cao, Zhuangying Xu, Fan Li, Wentao Zhong, Ke Huo, and Karthik Ramani. 2019. V.Ra: An In-Situ Visual Authoring System for Robot-IoT Task Planning with Augmented Reality. In *Proceedings of the 2019 on Designing Interactive Systems Conference* (San Diego, CA, USA) *(DIS '19)*. Association for Computing Machinery, New York, NY, USA, 1059–1070. https://doi.org/10.1145/3322276.3322278

[11] Sonia Mary Chacko and Vikram Kapila. 2019. An Augmented Reality Interface for Human-Robot Interaction in Unconstrained Environments. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 3222–3228. https://doi.org/10.1109/IROS40897.2019.8967973

[12] Zhutian Chen, Wai Tong, Qianwen Wang, Benjamin Bach, and Huamin Qu. 2020. Augmenting Static Visualizations with PapARVis Designer. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3313831.3376436

[13] EunJeong Cheon, Eike Schneiders, Kristina Diekjobst, and Mikael B. Skov. 2022. Robots as a Place for Socializing: Influences of Collaborative Robots on Social Dynamics In- and Outside the Production Cells. *Proc. ACM Hum.-Comput. Interact.* 6, CSCW2, Article 457 (nov 2022), 26 pages. https://doi.org/10.1145/3555558

[14] Neil Chulpongsatorn, Mille Skovhus Lunding, Nishan Soni, and Ryo Suzuki. 2023. Augmented Math: Authoring AR-Based Explorable Explanations by Augmenting Static Math Textbooks. In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23)*. ACM. https://doi.org/10.1145/3586183.3606827

[15] Andre Cleaver, Darren Vincent Tang, Victoria Chen, Elaine Schaertl Short, and Jivko Sinapov. 2021. Dynamic Path Visualization for Human-Robot Collaboration. In *Companion of the 2021 ACM/IEEE International Conference on Human-Robot Interaction* (Boulder, CO, USA) *(HRI '21 Companion)*. Association for Computing Machinery, New York, NY, USA, 339–343. https://doi.org/10.1145/3434074.3447188

[16] Yunus Emre Cogurcu and Steve Maddock. 2023. Augmented Reality Safety Zone Configurations in Human-Robot Collaboration: A User Study. In *Companion of the 2023 ACM/IEEE International Conference on Human-Robot Interaction* (Stockholm, Sweden) *(HRI '23)*. Association for Computing Machinery, New York, NY, USA, 360–363. https://doi.org/10.1145/3568294.3580106

[17] Anca D. Dragan, Shira Bauman, Jodi Forlizzi, and Siddhartha S. Srinivasa. 2015. Effects of Robot Motion on Human-Robot Collaboration. In *Proceedings of the Tenth Annual ACM/IEEE International Conference on Human-Robot Interaction* (Portland, Oregon, USA) *(HRI '15)*. Association for Computing Machinery, New York, NY, USA, 51–58. https://doi.org/10.1145/2696454.2696473

[18] Miguel Faria, Andrea Costigliola, Patrícia Alves-Oliveira, and Ana Paiva. 2016. Follow me: Communicating intentions with a spherical robot. In *2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. 664–669. https://doi.org/10.1109/ROMAN.2016.7745189

[19] Anna Fuste, Ben Reynolds, James Hobin, and Valentin Heun. 2020. Kinetic AR: A Framework for Robotic Motion Systems in Spatial Computing. In *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI EA '20)*. Association for Computing Machinery, New York, NY, USA, 1–8. https://doi.org/10.1145/3334480.3382814

[20] Ramsundar Ganesan, Yash Rathore, Heather Ross, and Heni Ben Amor. 2018. Better Teaming Through Visual Cues: How Projecting Imagery in a Workspace Can Improve Human–Robot Collaboration. *IEEE Robotics and Automation Magazine* PP (05 2018), 1–1. https://doi.org/10.1109/MRA.2018.2815655

[21] Michael J. Gielniak and Andrea L. Thomaz. 2011. Generating anticipation in robot motion. In *2011 RO-MAN*. 449–454. https://doi.org/10.1109/ROMAN.2011.6005255

[22] Terrell Glenn, Ananya Ipsita, Caleb Carithers, Kylie Peppler, and Karthik Ramani. 2020. StoryMakAR: Bringing Stories to Life With An Augmented Reality & Physical Prototyping Toolkit for Youth. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–14. https://doi.org/10.1145/3313831.3376790

[23] Uwe Gruenefeld, Lars Prädel, Jannike Illing, Tim Stratmann, Sandra Drolshagen, and Max Pfingsthorn. 2020. Mind the ARm: Realtime Visualization of Robot Motion Intent in Head-mounted Augmented Reality. In *Mensch und Computer 2020*. https://doi.org/10.1145/3404983.3405509

[24] Dave Hershberger, David Gossow, and Josh Faust. 2019. RViz, 3D visualization tool for ROS. http://wiki.ros.org/rviz. (Accessed on 23/12/2023).

[25] Nicholas J. Hetherington, Elizabeth A. Croft, and H.F. Machiel Van der Loos. 2021. Hey Robot, Which Way Are You Going? Nonverbal Motion Legibility Cues for Human-Robot Spatial Interaction. *IEEE Robotics and Automation Letters* 6, 3 (July 2021), 5010–5015. https://doi.org/10.1109/LRA.2021.3068708

[26] Antti Hietanen, Roel Pieters, Minna Lanz, Jyrki Latokartano, and Joni-Kristian Kämäräinen. 2020. AR-based interaction for human-robot collaborative manufacturing. *Robotics and Computer-Integrated Manufacturing* 63 (2020), 101891. https://doi.org/10.1016/j.rcim.2019.101891

[27] Khoa C. Hoang, Wesley P. Chan, Steven Lay, Akansel Cosgun, and Elizabeth A. Croft. 2022. ARviz: An Augmented Reality-Enabled Visualization Platform for ROS Applications. *IEEE Robotics & Automation Magazine* 29, 1 (2022), 58–67. https://doi.org/10.1109/MRA.2021.3135760

[28] Bryce Ikeda and Daniel Szafir. 2022. Advancing the Design of Visual Debugging Tools for Roboticists. In *2022 17th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 195–204. https://doi.org/10.1109/hri53351.2022.9889392

[29] ISO Central Secretary. 2016. *Robots and robotic devices — Collaborative robots*. Standard ISO/TS 15066:2016. International Organization for Standardization. https://www.iso.org/standard/62996.html

[30] Veronika Krauß, Michael Nebeling, Florian Jasche, and Alexander Boden. 2022. Elements of XR Prototyping: Characterizing the Role and Use of Prototypes in Augmented and Virtual Reality Design. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 310, 18 pages. https://doi.org/10.1145/3491102.3517714

[31] David Ledo, Steven Houben, Jo Vermeulen, Nicolai Marquardt, Lora Oehlberg, and Saul Greenberg. 2018. Evaluation Strategies for HCI Toolkit Research. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) *(CHI '18)*. Association for Computing Machinery, New York, NY, USA, 1–17. https://doi.org/10.1145/3173574.3173610

[32] Germán Leiva, Jens Emil Grønbæk, Clemens Nylandsted Klokmose, Cuong Nguyen, Rubaiat Habib Kazi, and Paul Asente. 2021. *Rapido: Prototyping Interactive AR Experiences through Programming by Demonstration.* Association for Computing Machinery, New York, NY, USA, 626–637. https://doi.org/10.1145/3472749.3474774

[33] Germán Leiva, Cuong Nguyen, Rubaiat Habib Kazi, and Paul Asente. 2020. Pronto: Rapid Augmented Reality Video Prototyping Using Sketches and Enaction. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3313831.3376160

[34] Gregory Lemasurier, Gal Bejerano, Victoria Albanese, Jenna Parrillo, Holly A. Yanco, Nicholas Amerson, Rebecca Hetrick, and Elizabeth Phillips. 2021. Methods for Expressing Robot Intent for Human–Robot Collaboration in Shared Workspaces. *J. Hum.-Robot Interact.* 10, 4, Article 40 (sep 2021), 27 pages. https://doi.org/10.1145/3472223

[35] Mille Skovhus Lunding, Jens Emil Sloth Grønbæk, Karl-Emil Kjær Bilstrup, Marie-Louise Stisen Kjerstein Sørensen, and Marianne Graves Petersen. 2022. ExposAR: Bringing Augmented Reality to the Computational Thinking Agenda through a Collaborative Authoring Tool. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 131, 14 pages. https://doi.org/10.1145/3491102.3517636

[36] Rasmus Lunding, Mathias Lystbæk, Tiare Feuchtner, and Kaj Grønbæk. 2023. AR-supported Human-Robot Collaboration: Facilitating Workspace Awareness and Parallelised Assembly Tasks. In *2023 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. 1064–1073. https://doi.org/10.1109/ISMAR59233.2023.00123

[37] Sotiris Makris, Panagiotis Karagiannis, Spyridon Koukas, and Aleksandros-Stereos Matthaiakis. 2016. Augmented reality system for operator support in human–robot collaborative assembly. *CIRP Annals* 65, 1 (2016), 61–64. https://doi.org/10.1016/j.cirp.2016.04.038

[38] Ali Ahmad Malik and Vijitashwa Pandey. 2022. Drive the Cobots Aright: Guidelines for Industrial Application of Cobots. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Vol. 86250. American Society of Mechanical Engineers. https://doi.org/10.1115/detc2022-90777 pubmed:ASME 2022 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference

[39] Joseph E. Michaelis, Amanda Siebert-Evenstone, David Williamson Shaffer, and Bilge Mutlu. 2020. Collaborative or Simply Uncaged? Understanding Human-Cobot Interactions in Automation. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association

for Computing Machinery, New York, NY, USA, 1–12. https://doi.org/10.1145/3313831.3376547

[40] AJung Moon, Daniel M. Troniak, Brian Gleeson, Matthew K.X.J. Pan, Minhua Zheng, Benjamin A. Blumer, Karon MacLean, and Elizabeth A. Croft. 2014. Meet Me Where i'm Gazing: How Shared Attention Gaze Affects Human-Robot Handover Timing. In *Proceedings of the 2014 ACM/IEEE International Conference on Human-Robot Interaction* (Bielefeld, Germany) *(HRI '14)*. Association for Computing Machinery, New York, NY, USA, 334–341. https://doi.org/10.1145/2559636.2559656

[41] Ana Moya, Leire Bastida, Pablo Aguirrezabal, Matteo Pantano, and Patricia Abril-Jiménez. 2023. Augmented Reality for Supporting Workers in Human & Robot Collaboration. *Multimodal Technologies and Interaction* 7, 4 (2023). https://doi.org/10.3390/mti7040040

[42] James F. Mullen, Josh Mosier, Sounak Chakrabarti, Anqi Chen, Tyler White, and Dylan P. Losey. 2021. Communicating Inferred Goals With Passive Augmented Reality and Active Haptic Feedback. *IEEE Robotics and Automation Letters* 6, 4 (2021), 8522–8529. https://doi.org/10.1109/LRA.2021.3111055

[43] Michael Nebeling and Katy Madier. 2019. 360proto: Making Interactive Virtual Reality and Augmented Reality Prototypes from Paper. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) *(CHI '19)*. Association for Computing Machinery, New York, NY, USA, 1–13. https://doi.org/10.1145/3290605.3300826

[44] Max Pascher, Uwe Gruenefeld, Stefan Schneegass, and Jens Gerken. 2023. How to Communicate Robot Motion Intent: A Scoping Review, In Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems. *arXiv preprint arXiv:2303.00362* 409, 17. https://doi.org/10.1145/3544548.3580857

[45] David J. Porfirio, Laura Stegner, Maya Cakmak, Allison Sauppe, Aws Albarghouthi, and Bilge Mutlu. 2021. Figaro: A Tabletop Authoring Environment for Human-Robot Interaction. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) *(CHI '21)*. Association for Computing Machinery, New York, NY, USA, Article 414, 15 pages. https://doi.org/10.1145/3411764.3446864

[46] Iulian Radu and Blair MacIntyre. 2009. Augmented-Reality Scratch: A Children's Authoring Environment for Augmented-Reality Experiences. In *Proceedings of the 8th International Conference on Interaction Design and Children* (Como, Italy) *(IDC '09)*. Association for Computing Machinery, New York, NY, USA, 210–213. https://doi.org/10.1145/1551788.1551831

[47] Eric Rosen, David Whitney, Elizabeth Phillips, Gary Chien, James Tompkin, George Konidaris, and Stefanie Tellex. 2019. Communicating Robot Arm Motion Intent Through Mixed Reality Head-Mounted Displays. *The International Journal of Robotics Research* 38, 12-13 (01 2019), 1513–1526. https://doi.org/10.1177/0278364919842925

[48] Ryo Suzuki, Adnan Karim, Tian Xia, Hooman Hedayati, and Nicolai Marquardt. 2022. 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* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 553, 33 pages. https://doi.org/10.1145/3491102.3517719

[49] Gilbert Tang, Phil Webb, and John Thrower. 2019. The development and evaluation of Robot Light Skin: A novel robot signalling system to improve communication in industrial human–robot collaboration. *Robotics and Computer-Integrated Manufacturing* 56 (2019), 85–94. https://doi.org/10.1016/j.rcim.2018.08.005

[50] Amanda Trang and Laurie Cheers. 2021. Introducing: Unity Robotics Visualizations Package. https://blog.unity.com/engine-platform/introducing-unity-robotics-visualizations-package. (Accessed on 23/12/2023).

[51] Georgios Tsamis, Georgios Chantziaras, Dimitrios Giakoumis, Ioannis Kostavelis, Andreas Kargakos, Athanasios Tsakiris, and Dimitrios Tzovaras. 2021. Intuitive and Safe Interaction in Multi-User Human Robot Collaboration Environments through Augmented Reality Displays. In *2021 30th IEEE International Conference on Robot Human Interactive Communication (RO-MAN)*. IEEE Press, Vancouver, BC, Canada, 520–526. https://doi.org/10.1109/RO-MAN50785.2021.9515474

[52] Ana Villanueva, Zhengzhe Zhu, Ziyi Liu, Kylie Peppler, Thomas Redick, and Karthik Ramani. 2020. Meta-AR-App: An Authoring Platform for Collaborative Augmented Reality in STEM Classrooms. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) *(CHI '20)*. Association for Computing Machinery, New York, NY, USA, 1–14. https://doi.org/10.1145/3313831.3376146

[53] Tianyi Wang, Xun Qian, Fengming He, Xiyun Hu, Yuanzhi Cao, and Karthik Ramani. 2021. GesturAR: An Authoring System for Creating Freehand Interactive Augmented Reality Applications. In *The 34th Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) *(UIST '21)*. Association for Computing Machinery, New York, NY, USA, 552–567. https://doi.org/10.1145/3472749.3474769

[54] Tianyi Wang, Xun Qian, Fengming He, Xiyun Hu, Ke Huo, Yuanzhi Cao, and Karthik Ramani. 2020. CAPturAR: An Augmented Reality Tool for Authoring Human-Involved Context-Aware Applications. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) *(UIST '20)*. Association for Computing Machinery, New York, NY, USA, 328–341. https://doi.org/10.1145/3379337.3415815

[55] Hui Ye and Hongbo Fu. 2022. ProGesAR: Mobile AR Prototyping for Proxemic and Gestural Interactions with Real-World IoT Enhanced Spaces. In *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems* (New Orleans, LA, USA) *(CHI '22)*. Association for Computing Machinery, New York, NY, USA, Article 130, 14 pages. https://doi.org/10.1145/3491102.3517689

[56] Zhengzhe Zhu, Ziyi Liu, Tianyi Wang, Youyou Zhang, Xun Qian, Pashin Farsak Raja, Ana Villanueva, and Karthik Ramani. 2022. MechARspace: An Authoring System Enabling Bidirectional Binding of Augmented Reality with Toys in Real-Time. In *Proceedings of the 35th Annual ACM Symposium on User Interface Software and Technology* (Bend, OR, USA) *(UIST '22)*. Association for Computing Machinery, New York, NY, USA, Article 50, 16 pages. https://doi.org/10.1145/3526113.3545668

Received 29 September 2023