Robot Programming Through Augmented Trajectories

Camilo Perez Quintero
CARIS Lab, University of British Columbia

Sarah Li
Department of Aeronautics and Astronautics, University of Washington

Cole Shing
CARIS Lab University of British Columbia

Wesley Chan
CARIS Lab, University of British Columbia

Sara Sheikholeslami
CARIS Lab, University of British Columbia

H.F. Machiel Van der Loos
CARIS Lab, University of British Columbia

Elizabeth Croft
Monash University

ABSTRACT

This paper presents a future-focused approach for robot programming based on augmented trajectories. Using a mixed reality head-mounted display (Microsoft Hololens) and a 7-DOF robot arm, we designed three augmented reality (AR) interactions to ease the robot programming task. 1) Trajectory specification. 2) Virtual previews of the robot motion. 3) Visualization of robot variables and reprogramming capabilities during simulation or execution. As a proof of concept, we illustrate our AR manufacturing paradigm by interacting with a 7-DOF robot arm to reduce wrinkles during the pleating step of vacuum bagging carbon-fiber-reinforcement-polymer process in a simulated scenario.

ACM Reference Format:
Camilo Perez Quintero, Sarah Li, Cole Shing, Wesley Chan, Sara Sheikholeslami, H.F. Machiel Van der Loos, and Elizabeth Croft. 2018. Robot Programming Through Augmented Trajectories. In *Proceedings of Workshop submission (VAM-HRI)*. ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Manufacturing is evolving. Online and on-demand purchasing in a global marketplace, at both the consumer and business-to-business level, has forced modern manufacturers to deal with high product customization and variation, just-in-time production and minimal warehousing, shorter product life cycles, and smaller batch sizes [1]. Furthermore, the introduction of sophisticated manufacturing processes and new materials that require a human expert to contribute to different stages of the production have brought higher skilled workers to production lines. The next generation of robots working in factories is being designed to work and interact in a complementary fashion alongside skilled human workers, completing collaborative tasks that increase overall productivity.

This new paradigm in industrial manufacturing demands more flexible and intuitive ways of programming industrial robots that allow the user to focus on the task and not in the complexity of the robotic system [4, 7, 9]. Augmented Reality (AR) is a promising alternative for the industry; It facilitates interaction by enhancing the physical world with virtual computer-generated information [2]. Assembly [14], training [15], maintenance [5] and repair [8] have been some of the industrial activities that researchers have explored using AR. Merging AR with robotic systems bring new Human-Robot Interaction (HRI) possibilities. An essential component for an effective HRI is the quality of the shared space between human and robot [3]. In traditional robot programming, the shared space is limited to setting way-points with a teach pendant and then replay the movement through them. AR offers a new communication channel to enhance the shared space. For instance, Fang *et al.* [6] prosed an AR interface that uses a marker-cube attached to a probe that allows the user to guide a virtual robot by setting way-points and orientations. The AR scene was visualized through a monitor. The authors pointed out that participants tended to focus on the screen more than on the real scenario and because depth is difficult to perceive from the display, it results into the improper guidance of the robot.

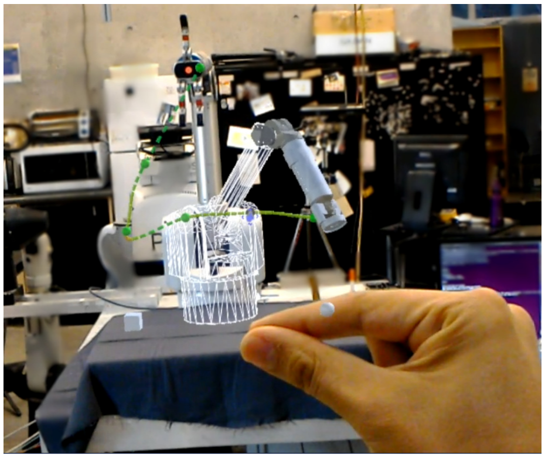

Figure 1: Hololens user point of view: The user creates, modifies, simulates and executes a pick-and-place trajectory using our AR robotic system. Notice that the holographic 7-DOF robot arm overlays the real robot and permits the user to simulate motions before actual execution. All the interaction is performed through speech and gestures.

VAM-HRI, March 2018, Chicago, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM. . . $0.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

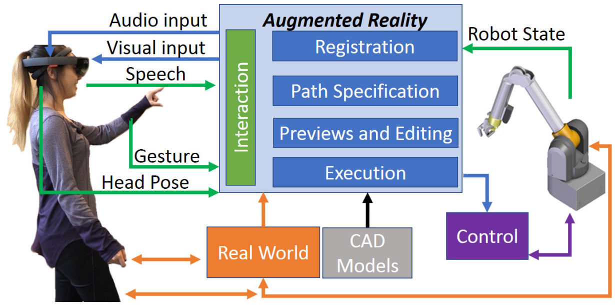

Figure 2: System Block Diagram

We aim to explore through Augmented Reality (AR), capabilities such as in-situ task specification, simulation, visualization and online interaction during the task. In this paper, we proposed an augmented reality (AR) manufacturing paradigm through augmented trajectories in which the user can: 1) Specify on the fly a robot trajectory on free space or in contact with a surface. 2) Visualize a preview of the robot movement. 3) Monitor and modify robot variables during the simulation or execution mode. We provide a preliminary evaluation of our AR robotic system by completing a labor intense step during carbon-fiber-reinforcement-polymer manufacturing. Based on the experiment, we provide guidelines and insights for creating human-robot interactive workspaces through augmenting reality.

2 SYSTEM

Trajectory programming is a core task in industrial robot automation. However, traditional methods are tedious, non-intuitive and visually non-collocated with the user's intended task. Our research tries to overcome those limitations by generalizing trajectory specification in-situ by providing two programming modes, free space trajectories, and surface trajectories. Figure 3 and 4 illustrate the two modes respectively and show some functionalities of our AR robotic system. In the former mode, a pick and place task is completed. The user wears a Hololens and through speech and gesture input is capable of select pick and place locations in the environment and then have in-situ previews of the robot movement. Furthermore, the user can modify the path dynamically in free space during simulation and execution. In the second example, Figure 4, a user is asked to erase a region of a white-board. Using head tracking, speech recognition and gesture detection the user set a virtual path on the surface and then interactively by using a MYO armband [13] as a 1DOF control input, the user controls the end-effector through the path keeping a normal orientation to the surface and applying a constant normal force. In the following section, we provide details on how these functionalities were implemented[1].

2.1 System pipeline

Our AR robotic system is composed of a back-drivable torque control 7-DOF Barrett Whole-Arm Manipulator (WAM) and the Microsoft Hololens, which is the first untethered mixed reality head-mounted display that allows holographic visualization [10]. Our general robot programming pipeline is shown in Figure 2. It consists of six modules: interaction, registration, path specification, preview and editing, execution and control.

The ***interaction module*** builds on top of the Hololens SDK gesture and speech recognition libraries and provides the user interactions such as editing a virtual path by pinch gestures, multiple speech commands to change state during the interaction, a virtual cursor controlled by the user's head pose that moves over holograms, etc.

The ***registration module*** registers the virtual and real world. This can be done through AR markers, by manually matching a virtual and real reference (Figure 3D) or by using the Hololen's spatial scanning capabilities and then place holograms on the scan surface, e.g., in Figure 3B the robot hologram is located within the real robot and in Figure 4D the user found easier to interact with the robot by having its hologram on its side.

The ***path specification module*** implements two types of paths, surface path or free space path. In the former, path points are set by using a virtual cursor that is visualized on the intersection between the 3D mesh or CAD model and a ray coming from the user's head orientation. After the user localizes the cursor in a point location, a speech command anchors the position to the 3D surface (Figure 4A and B). The path is defined by multiple points. In the free space path case, the user defines two anchor points (cube and sphere in Figure 3B) and a path is generated between the points and the robot's end-effector (Figure 3C).

The path is calculate base on the control points $x_1, ..., x_n$ using B-splines [11] with degree $k = 3$ of the form:

$$B_i^k(x) = v_i^k(x) B_i^{k-1}(x) + \left(1 - v_{i+1}^k(x)\right) B_{i+1}^{k-1}(x) \tag{1}$$

Where,

$$v_i^k(x) = \frac{x - x_i}{x_{i+k} - x_i} \quad \text{and} \quad B_i^0(x) = \begin{cases} 1 & \text{if } x_i \le x < x_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

The motivation of using B-splines instead of standard polynomial representation is that our implementation allows dynamical path specification, in the polynomial case changing any point requires a change in all coefficients of the spline interpolation while in the B-splines case only a few basis functions are affected.

In the ***preview and editing module***, a virtual simulation of the robot movement is trigger by speech. The current joint state of the real robot is used as initial position and then based on the virtual path, robot trajectory planning calculations are done and visualized. During the simulation, the path and robot planning can be dynamically changed using a pinch gesture (Figure 3C).

In the ***execution module***, the robot movement is trigger by speech command. This execution can be done autonomously or semi-autonomously depending on the process and the user preference.

Based on user proximity and if the task requires contact with the environment, the ***control module*** will operate on a position base controller mode or in an impedance controller mode. Our impedance controller is based on previous work [12]. A contact path interaction is shown in Figure 4. The user sets the path points by orienting a virtual cursor with his head and anchor the point by using a speech command (Figure 4A and B), the corresponding normal is calculated base on the 3D environmental reconstruction or in the CAD model. Then, the user by a speech command locks the robot to the path. The robot controller utilizes the set points $\boldsymbol{x}_1, ..., \boldsymbol{x}_n$ and its normals (blue

---

[1] Video demonstration of our system – https://youtu.be/viOTlzEv_Rw

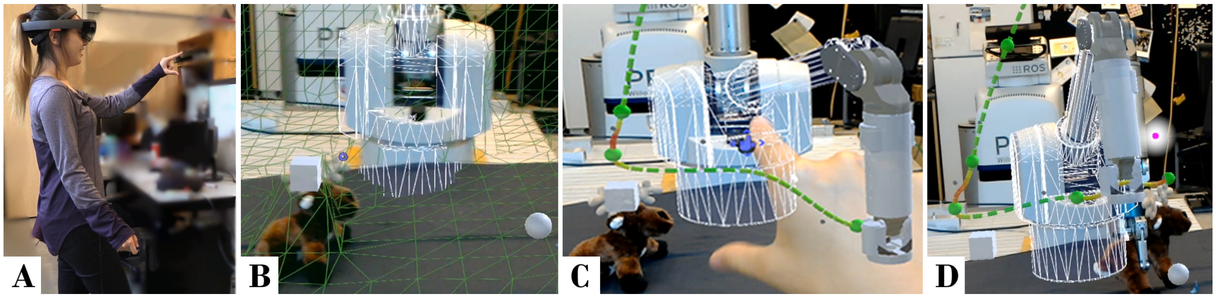

Figure 3: A pick and place task is completed using our AR Robotics system. (A) An operator wearing the Hololens and gesturing to interact with a 7DOF robot arm. (B) Visualization of, 3D spatial grid, real and virtual robot arm, pick (grey cube) and place (grey sphere) location. (C) Editing visual trajectory by gesturing and then simulate virtual robot through the trajectory. (D) Pick and place execution with virtual and real robot overlapping.

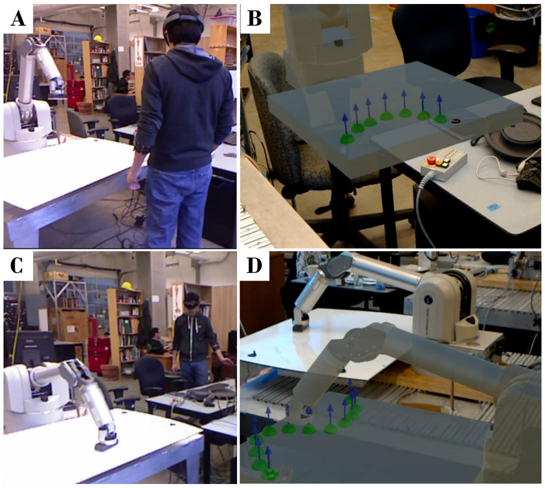

Figure 4: (A) The user set the virtual robot and the CAD model of the whiteboard in the side of the real robot. (B) The user set the path points on the virtual surface. (D) The user constrain the end-effector of the robot to the path with an opposite orientation to the surface normals. (C) The user controls the robot movements inside the path by waving his hand using a MYO device.

arrow Figure 4A and D) to constrain the robot end-effector to the path by minimize the projection of the distance vector between the robot's end-effector and the closest point in the path onto $\hat{\boldsymbol{s}} = \hat{\boldsymbol{n}} \times \hat{\boldsymbol{t}}$ direction. Where $\hat{\boldsymbol{n}}$ is the normal unit vector and $\hat{\boldsymbol{t}}$ is the tangential unit vector calculate from the 3D point path $\hat{\boldsymbol{t}}_i = \boldsymbol{x}_{d_i} - \boldsymbol{x}_{d_{i-1}}$. The force controller is given by,

$$F_s = K_{p_p}\left((\boldsymbol{x} - \boldsymbol{x}_d) \cdot \hat{\boldsymbol{s}}\right) + K_{d_p}(\dot{\boldsymbol{x}} \cdot \hat{\boldsymbol{s}}) \tag{2}$$

as the magnitude of the side direction force, due to the virtual spring K_{p_p} and damper K_{d_p} system that pulls $\boldsymbol{x}$ towards the closest point on the reference path. The torque command that generates this force is

$$\boldsymbol{\tau}_p = \boldsymbol{J}^T(F_s\, \hat{\boldsymbol{s}}) \tag{3}$$

where $\boldsymbol{J}$ is the task Jacobian matrix of robot.

3 CASE STUDY

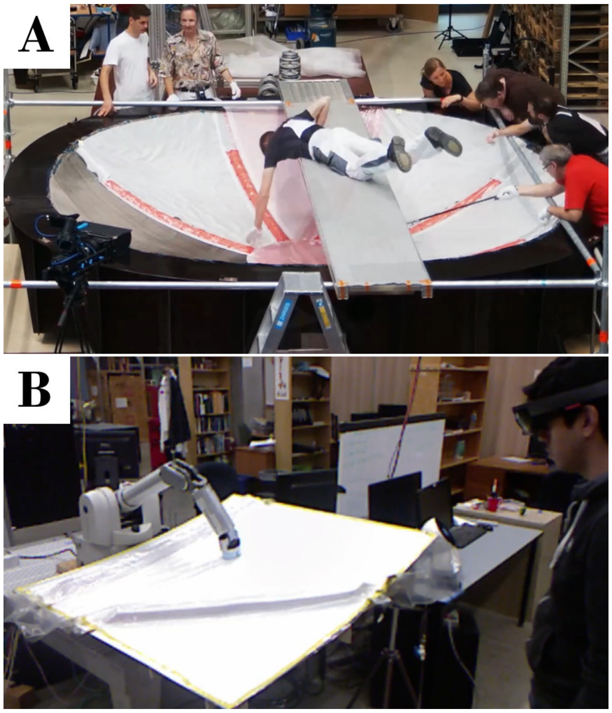

Figure 5: (A) Expert working on the manufacturing of a CFRP Pressure bulkhead, image courtesy of DLR. (B) Prototype of a section of the bulkhead mold.

A prime example of complex manufacturing is the production of carbon-fiber-reinforced-polymer (CFRP) components for the Aerospace industry. One method of producing CFRP parts is vacuum bagging. It consists of laying different carbon fiber cloths on a mold (Figure 5A) and then cover it with a plastic film and create a vacuum to subsequently diffuse resin into it. A previous step to the resin infusion is forming pleats that are needed to reduce air pockets which can lead to manufacturing defects such as bridging. This step involves dexterous interaction by an expert who provides

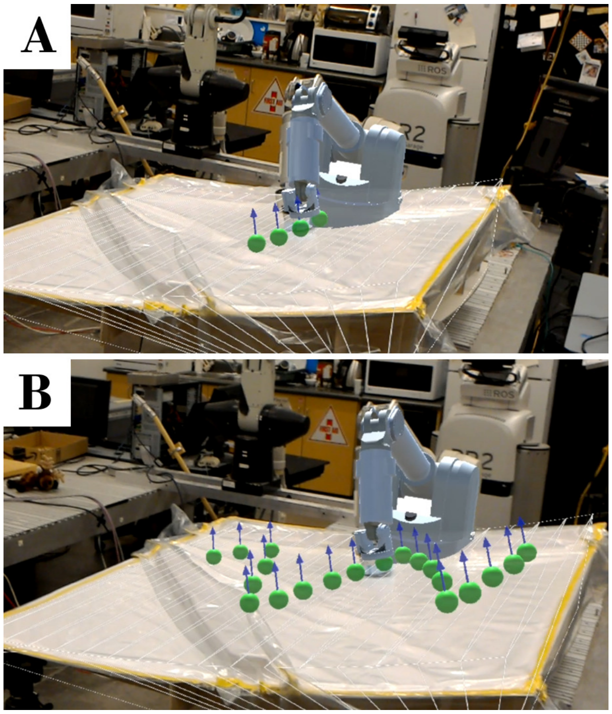

Figure 6: Two different sample strategies to reduce wrinkles in the plastic film. The virtual robot and the mold are overlaid on the real scenario. The end-effector is constrained to the path and its orientation is keep normal to the model surface. The user move the robot through a 1-DOF input device.

manual manipulation for identifying and handling the wrinkling and pleating of flexible membranes during manufacturing. This task is labor-intense task that requires an "online expert" is complicated to automate and currently is performed manually.

Our goal is to use our propose AR robotics interactions to assist the expert during this process. To this end, we have prototyped a section of a pressure bulkhead mold (Figure 5B) including the structural reinforcements. We ran a preliminary pilot study with one participant. We fitted our 7-DOF WAM arm with a round pad that guarantees friction with the plastic material. The participant task was to use our AR robotic system to move wrinkles to the main pleats.

To specify a path using our AR interface we first need to find a virtual surface representation of the working object. Our system support two modalities to integrate physical objects as holograms. The first approach leverages a spatial mapping. However, the plastic film presents challenges for the Hololens depth sensor; in this particular case, the spatial mapping will result in a poor 3D environmental reconstruction. A second approach consists of register a CAD model of the physical object to the physical environment. Figure 6 shows the users point of view through the Hololens. The CAD model of the section mold is visualized as a wire-frame. This translucid type of visualization is designed on purpose to allow the user to see in detail the plastic film wrinkles and base on that specify a path to reduce them. The normals on the selected path are calculated using the CAD model geometry and utilized to orient the end-effector while the movement is executed. The robot motion can be set automatically or control by the user through a 1-DOF input device. In our implementation, a MYO armband was used.

3.1 Experiments and Discussion

At the beginning of the experiment, the participant was instructed on the system functionality. Then, he was allowed to interact with the system for 15 minutes on a flat surface. Next, the section mold was placed close to the robot arm and register with the Hololens (Figure 6B). The user was asked to move all the visible wrinkles to the two main pleats located near the structural reinforcements. After completing the task, the plastic bag setup was reset. The participant repeats the test for six times. In all the six iterations the participant was able to remove the majority of the wrinkles inside the robot workspace. Two examples of strategies used during the test are shown in Figure 6A and B. In the former; the participant opts for specifying multiple paths perpendicular to the pleats. In the latter, the strategy was a single path that tries to move all the wrinkles to the main pleats. The second strategy was faster than the first one, but the wrinkles were reduced better using the first strategy. After the experiment, the participant was asked to complete a subjective questionnaire.

After running the experiment and receiving the user's qualitative feedback, the lessons learned are:

- The current implementation only allows setting points one by one on the surface. It would be worth to explore a faster way of setting multiple locations by recording the cursor trajectory.
- The calibration between the holograms and the real world stays accurate if there is not a significant displacement of the user, e.g., by moving behind the robot, the hologram looks offset about 3-5cm. In the experiment, we only used the 3D reconstruction performed by the depth sensor as the anchor method for the holograms. Potential improvement in the error can be achieved by using AR markers and fuse it with the calibration coming from the Hololens spatial reconstruction.
- The force applied to the surface is constant. Adding a safe way to interactively change the normal force to the surface, will improved the quality of the task.
- Although the plastic wrinkle configuration is hard to predict, executing predefined movements before starting the interactive wrinkle reduction, may reduce working time.

4 CONCLUSIONS AND FUTURE WORK

In this paper, we presented an AR robotic system for trajectory interaction, and through it, we exemplified two robot programming modalities, free space trajectory, and contact surface trajectory. We validate our system functionalities by completing a labor-intensive step during CFRP manufacturing. Potential directions of improvement are force visualization and control during the interaction, a more agile way of path specification, minimization of the registration error after user's long displacements.

REFERENCES

[1] Azin Aryania, Balazs Daniel, Trygve Thomessen, and Gabor Sziebig. 2012. New trends in industrial robot controller user interfaces. In *Cognitive Infocommunications (CogInfoCom), 2012 IEEE 3rd International Conference on*. IEEE, 365–369.

[2] Ronald T Azuma. 1997. A survey of augmented reality. *Presence: Teleoperators and virtual environments* 6, 4 (1997), 355–385.

[3] Cynthia Breazeal, Aaron Edsinger, Paul Fitzpatrick, and Brian Scassellati. 2001. Active vision for sociable robots. *IEEE Transactions on systems, man, and cybernetics-part A: Systems and Humans* 31, 5 (2001), 443–453.

[4] Torgny Brogårdh. 2007. Present and future robot control development- An industrial perspective. *Annual Reviews in Control* 31, 1 (2007), 69–79.

[5] Timo Engelke, Jens Keil, Pavel Rojtberg, Folker Wientapper, Michael Schmitt, and Ulrich Bockholt. 2015. Content first: a concept for industrial augmented reality maintenance applications using mobile devices. In *Proceedings of the 6th ACM Multimedia Systems Conference*. ACM, 105–111.

[6] HC Fang, SK Ong, and AYC Nee. 2014. Novel AR-based interface for human-robot interaction and visualization. *Advances in Manufacturing* 2, 4 (2014), 275–288.

[7] Kelleher R Guerin, Colin Lea, Chris Paxton, and Gregory D Hager. 2015. A framework for end-user instruction of a robot assistant for manufacturing. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE, 6167–6174.

[8] Steven Henderson and Steven Feiner. 2011. Exploring the benefits of augmented reality documentation for maintenance and repair. *IEEE transactions on visualization and computer graphics* 17, 10 (2011), 1355–1368.

[9] Clint Heyer. 2010. Human-robot interaction and future industrial robotics applications. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*. IEEE, 4749–4754.

[10] Bernard C Kress and William J Cummings. 2017. 11-1: Invited Paper: Towards the Ultimate Mixed Reality Experience: HoloLens Display Architecture Choices. In *SID Symposium Digest of Technical Papers*, Vol. 48. Wiley Online Library, 127–131.

[11] T Heath Michael. 2002. Scientific computing: an introductory survey. *The McGraw-540 Hill Companies Inc.: New York, NY, USA* (2002).

[12] Camilo Perez Quintero, Masood Dehghan, Oscar Ramirez, Marcelo H Ang, and Martin Jagersand. 2017. Flexible virtual fixture interface for path specification in tele-manipulation. In *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 5363–5368.

[13] Seema Rawat, Somya Vats, and Praveen Kumar. 2016. Evaluating and Exploring the MYO ARMBAND. In *System Modeling & Advancement in Research Trends (SMART), International Conference*. IEEE, 115–120.

[14] Xiangyu Wang, SK Ong, and AYC Nee. 2016. A comprehensive survey of augmented reality assembly research. *Advances in Manufacturing* 4, 1 (2016), 1–22.

[15] Sabine Webel, Uli Bockholt, Timo Engelke, Nirit Gavish, Manuel Olbrich, and Carsten Preusche. 2013. An augmented reality training platform for assembly and maintenance skills. *Robotics and Autonomous Systems* 61, 4 (2013), 398–403.