A Multimodal System using Augmented Reality, Gestures, and Tactile Feedback for Robot Trajectory Programming and Execution

Wesley P. Chan[1]*, Camilo Perez Quintero[1]*, Matthew K. X. J. Pan[1], Maram Sakr[1], H.F. Machiel Van der Loos[1], and Elizabeth Croft[2]

Abstract— **Currently available interfaces for programming industrial robots, e.g., teach pendants and computer consoles, are often unintuitive, resulting in a slow and tedious process for teaching robot tasks. Kinesthetic teaching, i.e., teaching robot motions by placing the robot in a gravity compensated state and then moving the robot though the desired motions, provides an alternative for small robots for which safe interaction can be guaranteed. However for many large industrial robots physical interaction is not an option. Emerging augmented reality technology offers an alternative interface with the potential to make robotic programming faster, safer, and more intuitive. The use of augmented reality admits the presentation of large amounts of rich, visual, in-situ information. However, it may also overload the user's visual information capacity, or may not provide sufficient feedback regarding the state of the robot. With the addition of gestural control and tactile feedback to augmented reality, we propose a system that allows users to program and execute robot tasks in an efficient and intuitive manner, by providing relevant feedback through different channels to maximize clear communication of the task commands and outcomes.**

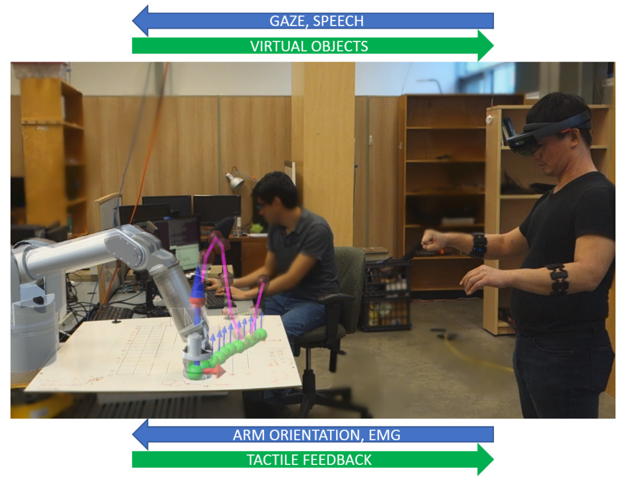

Fig. 1. A user showcases our multimodal system by following a sine force pattern (pink line). The user controls the normal force exerted on the surface (blue arrow) and the end effector linear velocity by moving his forearm and changing his muscle activation level.

I. INTRODUCTION

For decades, teach pendants, augmented with computer consoles, have been the de facto interface for programming industrial robots. Over time, this programming modality has seen little change, mainly due to the infrequency of robot programming for fully automated tasks once an assembly operation is set-up. However, the recent introduction of less-expensive and increasingly interactive robots has allowed for some flexibility in the manufacturing process. Thus, the infrequency of programming and reprogramming that was traditionally expected for industrial service robots may not apply. For example, kinesthetic teaching of robots such as Baxter, Sawyer, and the KUKA iiwa allows for easy and frequent reprogramming, permitting these robots to produce customized products in small lot sizes and thus able to generate a highly variable product mix. As a result, a paradigm shift has been observed where the industry is moving from large scale full automation, using only robots, to smaller scale reconfigurable robots and human-robot hybrid collaborative teams [1], [2], [3].

Along with this shift comes new requirements for flexible and intuitive methods to reprogram and interact with such industrial robots, in order to ensure safety and efficiency. Along with kinesthetic teaching methods, emerging augmented reality (AR) technology provides a promising alternative to traditional teach pendants for addressing such requirements. With increasing complexity of industrial robotic systems, which may not be safe for physical interaction, there is a growing demand for alternative user interfaces for robot programming. This alternative should provide sufficient capacity for communicating all the necessary information to the user, without adding a layer of complexity that distracts the user from the task [4], [5], [6]. Traditional programming methods lack such capacity, and often result in a cumbersome interaction. Augmented reality enables us to create a rich set of user interfaces that are co-located with the robot, allowing the user to have better situational awareness [7]. Furthermore, it permits visualization and interaction with hidden process variables that in traditional programming methods are not exposed to the operator during execution (e.g., force, velocity, acceleration). By improving the quality of shared information between human and robot, we can achieve more effective human-robot interaction [8].

With augmented reality devices and development tools such as the Microsoft HoloLens [9], Epson Moverio [10], and Magic Leap [11] increasingly available, researchers have explored the use of augmented reality for various tasks including assembly [12], maintenance [13], repair [14], and training [15] and found positive results. While augmented

[1]Collaborative Advanced Robotics and Intelligent Systems (CARIS) Laboratory, Department of Mechanical Engineering, University of British Columbia

[2]Monash University

*These authors contributed equally.

reality can provide a rich amount of visual information, too much information can cause perceptual overload [16]. Multimodal systems have been suggested to be more efficient because of its similarity with daily interactions [17]. Furthermore, not all task-relevant information is best communicated in a visual format. For example, higher-order motions (e.g., accelerations and jerks), as well as forces and moments (e.g., friction, torsion) can be difficult to visualize. When working on collaborative tasks, humans often utilize multiple inputs and communication channels including haptics [18], gestures [19], gaze [20], and posture [21]. Studies have shown that through a combination of communication channels for human-robot cooperation, we can achieve more effective interaction [22], [23]. In this work, we develop a robotic user interface system using augmented reality, gestural control, and haptic feedback for programing and controlling a robotic arm, and achieving intuitive and efficient collaboration between user and robot.

II. RELATED WORK

AR technology shows potential for improving human-robot interactions [24] and robot programming [25]. Many researchers have realized the potential of AR-based systems for robot path planning [26], controlling end effector orientation [27] and teleoperation [28]. Zaeh et al., [29] proposed an interactive AR-based interface for programming industrial robots. The virtual information relevant for programming, such as trajectories and target coordinates, is projected into the robots environment and can be manipulated interactively. A laser projection technique was used to allow the user to intuitively draw the desired motion path directly on the workpiece without requiring body-attached or handheld display devices. Their results showed that programming time reduced significantly by 80% compared to the conventional programming methods. Chong et al. [30] introduced a methodology for planning collision-free paths in AR environment. In addition, they proposed the usage of a scalable virtual robot which offers flexibility and adaptability to different environments when an in-situ robot programming approach is desired. Their system allowed the user to define the start, intermediate and end goal configurations for all paths. Then, beam search algorithm was used to generate the corresponding paths and the user had a simulated preview of the path before transferring it to the real robot. Green et al. [31] proposed using an AR system to teleoperate a remotely located mobile robot. They evaluated their system by comparing between a direct teleoperation interface, in which the user received visual cues from a camera mounted on top of the robot, and the AR system where the user has an exocentric view of the robot and allowed for gesture and speech interaction. They showed that the AR system had better accuracy for task completion whereas direct teleoperation was faster than the AR interface. In addition, their subjective questionnaire showed that users felt better situational awareness and the robot was more of a partner using the AR interface.

Despite these promising results, these AR systems lack force feedback since the objects created are only virtual. At the same time, force feedback during task execution is important and informative for achieving tasks successfully [32], [33], [34]. Force feedback can be provided visually through a virtual gauge or meter [35], or through an in-hand haptic device [36]. Using a visual gauge to provide force feedback may be unintuitive since we are mapping a haptic sense to the visual input. Further, using an in-hand haptic device will require the user to hold on to the device with one or both hands, removing the opportunity for the hand(s) to complete other work and, if the device is large or tethered, preventing the user from moving around the workspace to observe the work. To allow users to intuitively program robotic trajectories and execute force-relevant tasks successfully, we propose a hands-free, untethered system combining the use of augmented reality with gesture control and tactile and visual force feedback.

Specifically, in this paper we propose using multi-model force feedback to eliminate the limitations of single model usage. Many robotic tasks require the robot to move along a specified trajectory while applying a specific force profile along the path. Grinding, polishing, and welding are examples of such tasks. Here, we evaluate the usage of visual and haptic feedback to help the user maintain the predefined force profile while controlling the robot to move along a specific trajectory.

A. Haptics

Tactile displays offer users with an unobtrusive method of receiving information, and has been found to be particularly useful during mentally demanding contexts [37], [38], [39]. Using haptic feedback with, or in lieu of visual displays has been shown to potentially reduce overall mental workload relative to using only visual display of information [38], [40]. Several studies show that presenting tactile information also causes faster reaction time, and reduced perception of mental effort [41], [42].

Much of the early research in tactile displays has been oriented towards assisting users in navigating real and virtual environments [43], [41]. However, this has since expanded towards providing users timing cues [44] or even cues relating to the psycho-physiological state of the user [45]. With the rise of wearables and smartwatches within the past few years, the ubiquity of wearable vibrotactile interfaces have formatively transitioned wearable haptics from a niche research area into a commonplace feature smart devices.

Within the context of this work, our goal is to integrate haptics in a AR interaction system as a low-attention feedback mechanism to allow users more control over a human-robot interaction.

III. SYSTEM

A. Hardware

We built our system using the latest technologies developed in augmented reality and surface electromyography-based (sEMG) gesture recognition. Our system consists of two main user devices: the Microsoft HoloLens, which is the first untethered head-mounted augmented reality system [9], and the ThalmicLabs Myo armband, which is the first portable gesture-based input device using sEMG [46]. The HoloLens allows us to render 3D virtual objects on top of real objects in the physical world and provides speech recognition for audio input from the user. It also has an inertial measurement unit which can be used to track the user's gaze. The Myo armband has eight sets of electrodes and provides the muscle activation levels read from each electrode. It also provides recognition of six simple hand gestures as well as the arm's acceleration and orientation. The Myo armband has a vibrator embedded in it which can be used to provided tactile feedback to the user. In our current work, we use our system to control a 7 DOF Barrett Whole Arm Manipulator (WAM) (Figure 1).

B. Software

We used the Microsoft HoloLens SDK and Unity for developing our augmented reality application. Robot Operating System (ROS) is used for controlling and interfacing with the WAM and the Myo arm bands. The software package rosbridge is used for communication between the HoloLens and ROS controlling the WAM.

Our software system allows the user to visualize the robot, create a trajectory using gaze and speech commands, preview the trajectory, execute the trajectory using gestures, control the force profile of the trajectory online using muscle activation, and provides tactile and visual force feedback during execution. We describe each of these components in the following.

Robot visualization. Our system creates a kinematic model of the robot and renders the virtual robot on top of the real robot in physical space. As the person moves the backdrivable robot in gravity compensation mode, the joint angles of the real robot are sent to the virtual robot. The virtual robot then moves accordingly to match the joint angles of the real robot, providing an accurate visualization of the real robot to the user (Figure 2).

Trajectory creation. The user's gaze is tracked by tracking the user's head orientation. A ray is traced out in the gaze direction and its intersection with the environment is computed. The interesecting point is rendered as a red circular visual marker. The user creates a trajectory by looking at keypoints on the surface in front of the robot, and giving a speech command "set point" to set trajectory way points (Figure 2).

Trajectory preview. After each "set point" command is given, a green spherical marker with a blue normal arrow is created and rendered onto the surface to provide a visualization of the trajectory waypoints to the user. When the user gives a verbal "lock path" command, a trajectory is generated using the waypoints $x_1,\ldots,x_n$ with a 3rd degree B-splines [47] in the form of:

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

where,

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

Trajectory execution. During execution of the trajectory, the robot arm is constrained to the path using a force controller previously developed [48]. Given the end effector's current location $\boldsymbol{x}$ and the closest point on the path $\boldsymbol{x}_d$, a restoring force F_s is applied in the direction of $\hat{\boldsymbol{s}} = \hat{\boldsymbol{n}} \times \hat{\boldsymbol{t}}$, where $\hat{\boldsymbol{n}}$ is the normal unit vector and $\hat{\boldsymbol{t}}$ is the tangential unit vector of the path at point $\boldsymbol{x}_d$. F_s is calculated as

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

where (K_{p_p}) and (K_{d_p}) are the proportional and derivative gains.

The user controls the robot's movement along the path using the Myo arm band by moving the forearm left and right. The user beings trajectory control by holding the right arm out with the forearm parallel to the ground and making a fist gesture. When the fist gesture is first detected, the yaw orientation of the forearm θ_i is recorded. As the user moves the forearm, the yaw displacement is measured as $d\theta = \theta - \theta_i$, where θ is the current yaw angle measurement. A driving velocity proportional to $d\theta$ is then applied in the $\hat{\boldsymbol{t}}$ direction to move the end effector along the path.

Force control. The user controls the applied force along the $\hat{\boldsymbol{n}}$ direction by varying their muscle activation level. When the system initializes, the user is first asked to hold a right hand fist as hard as they can, and a maximum EMG reading $\bar{\varepsilon}_{max}$ is recorded. The average EMG activation level at time t is computed as:

$$\varepsilon_{avg}(t) = \sum_{i=1}^{8} \varepsilon_i(t) \tag{3}$$

where $\varepsilon_i(t)$ is the EMG reading of electrode i on the MYO band at time t. A smoothed average EMG signal is obtained by averaging $\varepsilon_i(t)$ over a time window:

$$\bar{\varepsilon}_{avg}(t) = \frac{\sum_i \varepsilon_{avg}(t-i)}{n} \tag{4}$$

The maximum EMG reading $\bar{\varepsilon}_{max}$ is taken as the $\bar{\varepsilon}_{avg}(t)$ value measured 3 seconds after the user was asked to hold a fist with maximum force to avoid initial spikes in the EMG reading. The force command is computed as:

$$F_n = F_{limit}\frac{\bar{\varepsilon}_{avg}(t) - \bar{\varepsilon}_{start}}{\bar{\varepsilon}_{max} - \bar{\varepsilon}_{start}} \tag{5}$$

where F_{limit} is set to 30N, and $\bar{\varepsilon}_{start}$ is the $\bar{\varepsilon}_{avg}$ value measured at the time when trajectory execution began (i.e., when the MYO band first detected the fist gesture).

Tactile force feedback. We created a tactile pattern feedback module for providing force feedback to the user via the vibrators on the MYO bands. The module allows arbitrary vibration patterns to be send to the arm of the user to signal various force events occurring at the robot end effector. Given the desired amount of force F_{target} at a point along the trajectory, and a tolerance δ, if the applied force $F_{applied}$ is lower than $F_{target} - \delta$, a vibration pattern of quick, short pulses with one second rest intervals is given. If $F_{applied}$ is higher than $F_{target} + \delta$, then a different vibration pattern of longer pulses with one second rest intervals is given. In our experiment, δ is set to 10N.

Visual force feedback. To be able to compare different feedback mechanisms, we also enabled our system to provide visual force feedback. When the end effector applies a force to the environment, arrows with lengths proportional to the applied force magnitudes are rendered at the end effector to provide a visualization of the forces in the xyz axes. These arrows travels with the end effector as it moves.

IV. EXPERIMENT

To demonstrate the usability of our system, we performed an experiment where a participant used our system to program and execute a trajectory with a given force profile. We tested three force profile trajectories with different complexity levels: a constant force profile, a ramp force profile, and sinusoidal force profile (Figure 2). For each of the force profiles, we also tested three modes of force feedback: tactile pattern only, visual display only, and tactile pattern with visual display.

In the experiment, the robot arm is located in front of a table with grid lines drawn on it, and the user stands across the robot on the other side of the table. In each trial, the user first creates a 2D spatial trajectory on the table by looking at key points on the table and speaking the command word "set point". After the key points have been set, the user then says "lock path" to generate a b-spline path through the key points and locking the robot's end effector to the generated path. The user then says "display force", and the desired force profile is displayed above the 2D trajectory. When executing each trajectory, the user is asked to first keep the robot arm at the starting point of the trajectory and attain the target force level. Once target force level at the starting point has been achieved, the user then proceeds to move the robot along the trajectory, while trying to follow the force profile. Once the robot has reached the end point of the trajectory, the trial is complete [TODO: add a series of figures showing setup and task execution].

V. ANALYSIS

For each trial, we computed the execution time, t_{exec}, the maximum absolute force error, $|e|_{max}$, the average absolute force error, $|e|_{avg}$, and the cumulative absolute force error, $I_{|e|}$. The execution time is defined as

$$t_{exec} = t_{end} - t_{start}, \tag{6}$$

where t_{start} is the time when the robot first moves more than 5cm away from the start point into the trajectory, and t_{end} is the time when the robot first moves past the end point of the trajectory. The maximum absolute force error is defined as

$$|e|_{max} = \max\{|e|(t) | t \in [t_{start}, t_{end}]\}, \tag{7}$$

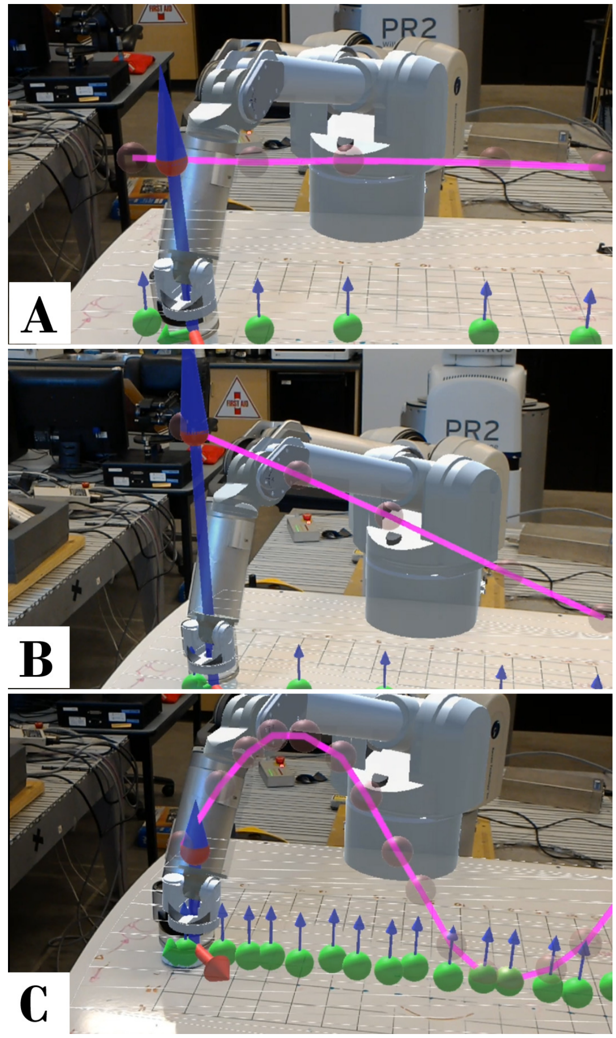

Fig. 2. Users Point of view. The pink line represents the force profile to follow: A. constant force. B. Ramp, C. Sine.

$$|e|(t) = |F_{applied}(t) - F_{target}|(t), \tag{8}$$

while the average absolute force error is defined as

$$|e|_{avg} = \frac{\sum\limits_{[t_{start}, t_{end}]} |e|(t)}{n}, \tag{9}$$

and the cumulative absolute force error is defined as

$$I_{|e|} = \sum_{[t_{start}, t_{end}]} |e|(t) * \Delta t, \tag{10}$$

VI. RESULT

The execution time, t_{exec}, maximum absolute force error, $|e|_{max}$, average absolute force error, $|e|_{avg}$, and cumulative absolute force error, $I_{|e|}$, measured in each trial, and the average, are shown in Table I, Table II, Table III, and Table IV respectively. Results show that in all conditions, the user was able to complete the trajectory within 17s. The average absolute force error was within the tolerance $\delta = 10N$ for all trials, and the maximum absolute force error did not exceed 20N. The tactile case has the smallest average $|e|_{max}$, $|e|_{avg}$,

TABLE I

EXECUTION TIME, t_{exec}, MEASURED IN EACH TRIAL.

t_{exec}(s)	Line	Ramp	Sin	Avg
Tactile	10.5	10.5	10.7	10.5
Visual	9.6	12.3	9.5	10.4
Tactile + Visual	11.3	13.2	16.9	13.8

TABLE II

MAXIMUM ABSOLUTE FORCE ERROR, $|e|_{max}$, MEASURED IN EACH TRIAL.

$\|e\|_{max}$(N)	Line	Ramp	Sin	Avg
Tactile	16.2	6.1	10.6	11.0
Visual	12.0	19.8	17.8	13.5
Tactile + Visual	10.3	12.0	15.5	12.6

and $I_{|e|}$ among all cases, while the visual case has the smallest average t_{exec}, with the t_{exec} measured in the tactile case only 0.1s longer. Second to the tactile case, the visual case has the next smallest $I_{|e|}$, performing better than the tactile and visual case, while the tactile and visual case has the next smallest $|e|_{max}$ and $|e|_{avg}$, performing better than the visual case. Figure 3 shows the plots of position and force error measured during the experiment, using the trial with tactile and visual feedback executing a sinusoidal force profile as an example.

VII. DISCUSSION

Our results showed that the user was able to complete all tasks within a reasonable amount of time and with the average force error kept below the chosen force error tolerance. Thus, our experiment has demonstrated the usability of our system. In addition, our data also revealed some noteworthy observations. While one might expect that providing both tactile and visual force feedback should allow the user to yield the best performance, results showed the contrary. Providing only tactile feedback yielded smaller force errors when compared to providing visual feedback or both tactile and visual feedback. A few possible explanations for these observations are as follows.

TABLE III

AVERAGE ABSOLUTE FORCE ERROR, $|e|_{avg}$, MEASURED IN EACH TRIAL.

$\|e\|_{avg}$(N)	Line	Ramp	Sin	Avg
Tactile	8.7	2.3	3.2	4.7
Visual	4.4	4.7	9.1	6.0
Tactile + Visual	5.4	4.4	7.0	5.6

TABLE IV

CUMULATIVE ABSOLUTE FORCE ERROR, $I_{|e|}$, MEASURED IN EACH TRIAL.

$I_{\|e\|}(N\dot{s})$	Line	Ramp	Sin	Avg
Tactile	68.9	24.3	33.9	42.4
Visual	41.8	46.4	86.7	58.3
Tactile + Visual	60.7	43.4	118.9	74.3

Force is a haptic sense. Thus, feeding it back to the user's haptic sensory input is a closer mapping than converting it to a form for the user's visual sensory input. As a result, a haptic feedback signal may be more easily processed and understood by the user's sensory and cognitive systems. Furthermore, the pathways through which haptic and visual inputs are processed by humans and converted into a resulting reaction can be different. A visual sensory signal needs to first travel to the brain and be processed by the brain before it is converted to a motor neuron signal and sent to the targeted muscle to generate a reaction. A haptic signal may travel through a different route known as the reflex arc, in which the input signal to the sensory neurons travels to the spinal cord, gets processed at the spinal cord, where a motor neuron signal is then sent out to generate the reaction, without the input signal needing to reach the brain for processing first [49]. The reflex arc is shorter than normal pathway through the brain, and thus, it has shorter reaction times. These could be explanations to why haptic feedback produces better performance than visual feedback.

A potential reason for why providing visual feedback in addition to haptic feedback did not improve the performance could be that the added visual feedback is imposing an additional load to the user's visual sensory input and demanding more cognitive processing from the user. The system already provides many other visual information through augmented reality related to trajectory creation and robot preview. Thus, adding visual force feedback may be overloading the visual channel.

VIII. CONCLUSION AND FUTURE WORK

We have created a system utilizing augmented reality, gesture control, and haptic feedback for intuitive and effective robotic trajectory programming and execution. To our knowledge, this is the first untethered, hands-free system providing online control and feedback for both trajectory and force. Our experiment demonstrated the usability of our system showing that the user was able to program and execute different force profile trajectories. Preliminary results

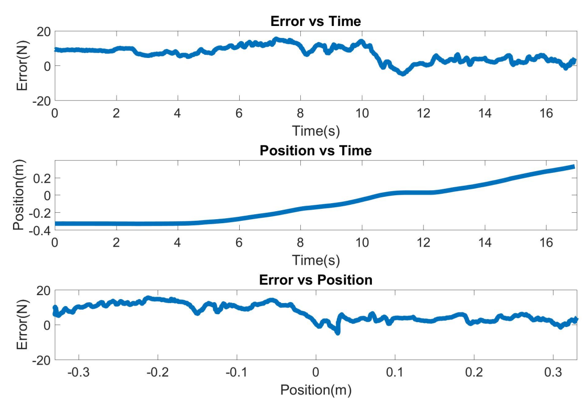

Fig. 3. Example experiment data from the trial with tactile and visual feedback executing a sinusoidal force profile.

showed favour for using haptic force feedback over visual force feedback.

In our next steps, we are planning on conducting larger user studies to collect more data for validating our preliminary results. In addition, we will be testing the different components in a systematic way, including comparing different methods for trajectory teaching, different forms of robotic input, and overall performance of the system on executing different tasks. Outcome of our work will guide the design of human robot interfaces utilizing the new technology of augmented reality towards achieving effective human robot collaboration.

References

[1] F. Wallhoff, J. Blume, A. Bannat, W. Rösel, C. Lenz, and A. Knoll, "A skill-based approach towards hybrid assembly," *Advanced Engineering Informatics*, vol. 24, no. 3, pp. 329–339, 2010.

[2] C. A. Moore, M. A. Peshkin, and J. E. Colgate, "Cobot Implementation fo Virtual Paths and 3-D Virtual Surfaces," vol. 19, no. 2, pp. 347–351, 2003.

[3] R. Wilcox, S. Nikolaidis, and J. Shah, "Optimization of Temporal Dynamics for Adaptive Human-Robot Interaction in Assembly Manufacturing," in *Robotics: Science and Systems*, 2012.

[4] K. R. Guerin, C. Lea, C. Paxton, and G. D. Hager, "A framework for end-user instruction of a robot assistant for manufacturing," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*. IEEE, 2015, pp. 6167–6174.

[5] C. Heyer, "Human-robot interaction and future industrial robotics applications," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*. IEEE, 2010, pp. 4749–4754.

[6] T. Brogårdh, "Present and future robot control development- an industrial perspective," *Annual Reviews in Control*, vol. 31, no. 1, pp. 69–79, 2007.

[7] R. T. Azuma, "A survey of augmented reality," *Presence: Teleoperators and virtual environments*, vol. 6, no. 4, pp. 355–385, 1997.

[8] C. Breazeal, A. Edsinger, P. Fitzpatrick, and B. Scassellati, "Active vision for sociable robots," *IEEE Transactions on systems, man, and cybernetics-part A: Systems and Humans*, vol. 31, no. 5, pp. 443–453, 2001.

[9] B. C. Kress and W. J. Cummings, "11-1: Invited paper: Towards the ultimate mixed reality experience: Hololens display architecture choices," in *SID Symposium Digest of Technical Papers*, vol. 48, no. 1. Wiley Online Library, 2017, pp. 127–131.

[10] "Augmented reality and mixed reality," 2018, https://epson.com/moverio-augmented-reality.

[11] "Magic leap," 2018, https://www.magicleap.com/.

[12] X. Wang, S. Ong, and A. Nee, "A comprehensive survey of augmented reality assembly research," *Advances in Manufacturing*, vol. 4, no. 1, pp. 1–22, 2016.

[13] T. Engelke, J. Keil, P. Rojtberg, F. Wientapper, M. Schmitt, and U. Bockholt, "Content first: a concept for industrial augmented reality maintenance applications using mobile devices," in *Proceedings of the 6th ACM Multimedia Systems Conference*. ACM, 2015, pp. 105–111.

[14] S. Henderson and S. Feiner, "Exploring the benefits of augmented reality documentation for maintenance and repair," *IEEE transactions on visualization and computer graphics*, vol. 17, no. 10, pp. 1355–1368, 2011.

[15] S. Webel, U. Bockholt, T. Engelke, N. Gavish, M. Olbrich, and C. Preusche, "An augmented reality training platform for assembly and maintenance skills," *Robotics and Autonomous Systems*, vol. 61, no. 4, pp. 398–403, 2013.

[16] R. Sigrist, G. Rauter, R. Riener, and P. Wolf, "Augmented visual, auditory, haptic, and multimodal feedback in motor learning: a review," *Psychonomic bulletin & review*, vol. 20, no. 1, pp. 21–53, 2013.

[17] P. Larsson, D. Vastfjall, and M. Kleiner, "Ecological acoustics and the multi-modal perception of rooms: real and unreal experiences of auditory-visual virtual environments." Georgia Institute of Technology, 2001.

[18] W. P. Chan, C. a. C. Parker, H. F. M. Van der Loos, and E. a. Croft, "A human-inspired object handover controller," *The International Journal of Robotics Research*, vol. 32, no. 8, pp. 971–983, 2013. [Online]. Available: http://ijr.sagepub.com/content/32/8/971.abstract

[19] A. Moon, C. A. C. Parker, E. A. Croft, and H. F. M. Van der Loos, "Did you see it hesitate? - Empirically grounded design of hesitation trajectories for collaborative robots," in *International Conference on Intelligent Robots and Systems*, 2011, pp. 1994–1999.

[20] M. Häring, J. Eichberg, and E. André, "Studies on Grounding with Gaze and Pointing Gestures in Human-Robot-Interaction," in *International Conference on Social Robotics*, 2012, pp. 378–387.

[21] M. Cakmak, S. S. Srinivasa, M. K. Lee, S. Kiesler, and J. Forlizzi, "Using Spatial and Temporal Contrast for Fluent Robot-Human Handovers," in *International Conference on Human Robot Interaction*, 2011, pp. 489–496.

[22] A. Moon, D. Troniak, and B. Gleeson, "Meet me where I'm gazing: how shared attention gaze affects human-robot handover timing," in *International Conference on Human-Robot Interaction*, 2014, pp. 334–341.

[23] A. Edsinger and C. C. Kemp, "Human-Robot Interaction for Cooperative Manipulation: Handing Objects to One Another," in *International Symposium on Robot and Human Interactive Communication*, 2007, pp. 1167–1172.

[24] S. A. Green, M. Billinghurst, X. Chen, and J. G. Chase, "Human-robot collaboration: A literature review and augmented reality approach in design," *International journal of advanced robotic systems*, vol. 5, no. 1, p. 1, 2008.

[25] Z. Pan, J. Polden, N. Larkin, S. Van Duin, and J. Norrish, "Recent progress on programming methods for industrial robots," *Robotics and Computer-Integrated Manufacturing*, vol. 28, no. 2, pp. 87–94, 2012.

[26] H. Fang, S. Ong, and A. Nee, "A novel augmented reality-based interface for robot path planning," *International Journal on Interactive Design and Manufacturing (IJIDeM)*, vol. 8, no. 1, pp. 33–42, 2014.

[27] ——, "Orientation planning of robot end-effector using augmented reality," *The International Journal of Advanced Manufacturing Technology*, vol. 67, no. 9-12, pp. 2033–2049, 2013.

[28] A. Yew, S. Ong, and A. Nee, "Immersive augmented reality environment for the teleoperation of maintenance robots," *Procedia CIRP*, vol. 61, pp. 305–310, 2017.

[29] M. F. Zaeh and W. Vogl, "Interactive laser-projection for programming industrial robots," in *Mixed and Augmented Reality, 2006. ISMAR 2006. IEEE/ACM International Symposium on*. IEEE, 2006, pp. 125–128.

[30] J. W. S. Chong, S. Ong, A. Y. Nee, and K. Youcef-Youmi, "Robot programming using augmented reality: An interactive method for planning collision-free paths," *Robotics and Computer-Integrated Manufacturing*, vol. 25, no. 3, pp. 689–701, 2009.

[31] S. A. Green, J. G. Chase, X. Chen, and M. Billinghurst, "Evaluating the augmented reality human-robot collaboration system," *International journal of intelligent systems technologies and applications*, vol. 8, no. 1-4, pp. 130–143, 2009.

[32] P. Kormushev, S. Calinon, and D. G. Caldwell, "Imitation learning of positional and force skills demonstrated via kinesthetic teaching and haptic input," *Advanced Robotics*, vol. 25, no. 5, pp. 581–603, 2011.

[33] F. Steinmetz, A. Montebelli, and V. Kyrki, "Simultaneous kinesthetic teaching of positional and force requirements for sequential in-contact tasks," *IEEE-RAS International Conference on Humanoid Robots*, vol. 2015-December, pp. 202–209, 2015.

[34] M. Minamoto, K. Kawashima, and T. Kanno, "Effect of force feedback on a bulldozer-type robot," in *2016 IEEE International Conference on Mechatronics and Automation, IEEE ICMA 2016*, 2016, pp. 2203–2208.

[35] A. Talasaz, A. L. Trejos, and R. V. Patel, "The Role of Direct and Visual Force Feedback in Suturing Using a 7-DOF Dual-Arm Teleoperated System," *IEEE Transactions on Haptics*, vol. 10, no. 2, pp. 276–287, 2017.

[36] B. Pier, P. Valentini, and M. E. Biancolini, "Interactive Sculpting Using Augmented- Reality, Mesh Morphing, and Force Feedback," no. march, pp. 83–90, 2018.

[37] A. Tang, P. Mclachlan, K. Lowe, C. R. Saka, and K. MacLean, "Perceiving Ordinal Data Haptically Under Workload," in *Proc. Int. Conf. Multimodal Interfaces*, Torento, 2005, pp. 317–324.

[38] B. M. Davis, "Effects of Visual, Auditory, and Tactile Navigation Cues on Navigation Performance, Situation Awareness, and Mental Workload, Tech. Rep. February, 2007.

[39] K. MacLean, "Foundations of Transparency in Tactile Information Design," *IEEE Transactions on Haptics*, vol. 1, no. 2, pp. 84–95, jul 2008. [Online]. Available: http://ieeexplore.ieee.org/document/4674348/
[40] K. Kahol, V. Hayward, and S. Brewster, "Ambient Haptic Systems," *IEEE Trans. Haptics*, vol. 2, no. 3, pp. 121–122, 2009.
[41] J. B. van Erp, H. A. van Veen, C. Jansen, and T. Dobbins, "Waypoint navigation with a vibrotactile waist belt," *ACM Trans. Applied Perception*, vol. 2, no. 2, pp. 106–117, apr 2005. [Online]. Available: http://dl.acm.org/citation.cfm?id=1060581.1060585
[42] S. Brewster and L. M. Brown, "Tactons: structured tactile messages for non-visual information display," in *Proc. AUIC*, vol. 4, 2004, pp. 15–23.
[43] S. Ertan, C. Lee, A. Willets, H. Tan, and A. Pentland, "A wearable haptic navigation guidance system," in *Int. Symp. Wearable Computers*. IEEE Comput. Soc, 1998, pp. 164–165. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=729547
[44] D. Tam, K. E. MacLean, J. McGrenere, and K. J. Kuchenbecker, "The design and field observation of a haptic notification system for timing awareness during oral presentations," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '13*. New York, New York, USA: ACM Press, 2013, p. 1689. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2470654.2466223
[45] M. K. X. J. Pan, J. McGrenere, E. A. Croft, and K. E. Maclean, "Exploring the Role of Haptic Feedback in an Implicit HCI-Based Bookmarking Application," *IEEE Transactions on Haptics*, vol. Submitted, pp. 1–12, 2012.
[46] "Thalmic labs," 2018, https://www.thalmic.com/.
[47] T. H. Michael, "Scientific computing: an introductory survey," *The McGraw-540 Hill Companies Inc.: New York, NY, USA*, 2002.
[48] C. P. Quintero, M. Dehghan, O. Ramirez, M. H. Ang, and M. Jagersand, "Flexible virtual fixture interface for path specification in telemanipulation," in *Robotics and Automation (ICRA), 2017 IEEE International Conference on*. IEEE, 2017, pp. 5363–5368.
[49] D. Kenneth S. Saladin, *Anatomy & Physiology: The Unity of Form and Function*. McGraw-Hill Education, 2014. [Online]. Available: https://books.google.ca/books?id=lmr8nQEACAAJ