Motion Constrained AR Control for Insertion Tasks

Dimitri Saliba
School of Robotics Engineering
Worcester Polytechnic Institute
Worcester, Massachusetts
dasaliba@wpi.edu

Antonio Sanchez
School of Robotics Engineering
Worcester Polytechnic Institute
Worcester, Massachusetts
asanchez@wpi.edu

Aman Gupta
School of Computer Science
Worcester Polytechnic Institute
Worcester, Massachusetts
agupta9@wpi.edu

Jaskrit Singh
School of Robotics Engineering
Worcester Polytechnic Institute
Worcester, Massachusetts
jsingh3@wpi.edu

***Abstract*—This paper proposes and evaluates a novel Augmented Reality (AR) interface for two human-robot interaction tasks. The interface focuses on improving the fine manipulation in insertion tasks by introducing motion constraints. The interface also facilitates the definition of complex polygons in 3D space for tasks such as wiping. We hypothesize that this interface will outperform traditional manual gesture control in terms of both task performance and user workload. To validate our hypothesis, we conducted a user study comparing the proposed interfaces with a conventional gesture-based control method. The study evaluated the proposed method via metrics such as completion time and accuracy, as well as subjective measures such as the NASA Task Load Index (TLX) to assess user experience and perceived workload. By analyzing the results of the user study, we demonstrate the advantages as well as limitations of our proposed methods.**

***Index Terms*—augmented reality, mixed reality, microsoft hololens 2**

I. Introduction

Augmented reality (AR) interfaces have become popular in the field of Human-Robot Interaction due to their ability to convey complex 3 dimensional information in an intuitive manner.

One of the most popular styles of AR interfaces is gesture control. This is primarily because gesture is intuitive, does not require physical remotes, and most AR devices (such as Microsoft HoloLens) include robust gesture detection libraries. However, despite its prevalence, gesture detection is prone to certain flaws regardless of implementation quality. One such flaw is the difficulty of fine manipulation. Since there is no haptic or proprioceptive feedback, gesture-based interfaces tend to have a feeling of weightlessness, which, combined with simultaneous control in all dimensions, can make fine manipulation such as alignment difficult.

We propose a motion-constrained gesture-based interface which addresses these issues by reducing the capacity for user error and emulating haptic feedback by restraining user input before collisions. In addition we propose a second AR interface for defining convex polygons in 3 dimensional space. This interface could be applied on its own for wiping tasks, or used to support the first interface, as a method for which a task operator can quickly define convex polygons about which to constrain motion.

Finally, our objective is to evaluate the efficacy of both interfaces against traditional gesture-based manual control through a user study. During the user study, we intend to evaluate both objective metrics, such as task performance, as well as subjective methods, such as the NASA-TLX survey in order to determine if our proposed interfaces outperform manual control in terms of performance and workload.

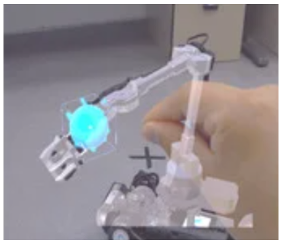
II. Related Works

A. Augmented Reality (AR) in Robotics

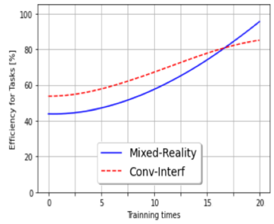
AR has emerged as a valuable tool in bridging the communication gap between humans and robots by overlaying interactive information over the real environment, enhancing human-robot interactions. A comprehensive survey of the field can be found in [1]. One of the key findings of the survey was that AR interfaces tend to cause a high cognitive load. This suggests that the key to good AR interface design is to present the most relevant information in the simplest way possible.

One paper [3] uses AR to control a mobile robot arm. Using the interface shown in 1a the user controls the arm through an AR sphere connected to the end effector. The robot then displays a hologram of the arm taking the desired motion allowing the user to double check the path before confirming the motion. Looking at the results from the paper in Figure 1b we can see that for low training time, the mixed reality (MR) interface is actually worse than a more conventional interface. It is only after the participants had around 16 training sessions that they became more efficient with the MR interfaces. This means that training time is an important variable to consider for MR/AR interfaces.

Another paper [8] focused on creating AR cues to help with situations where depth perception might be limited. They created two interfaces as show in Figure 2. The first was basic cues which you can see in Figure 2a. It shows a laser pointer coming down from the end effector with a circle at the bottom showing the grip radius. It also used color cues to explicitly



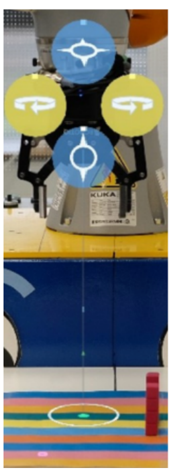
(a) Controlling robot arm using a virtual sphere. Robot displays hologram of target position.



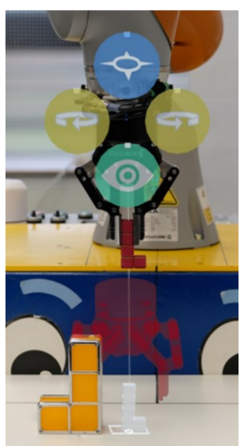
(b) Task Efficiency vs Training Time for a conventional and a mixed reality interface.

Fig. 1: Interface and results from [3]

show depth. The second interface assumes that 3-D models of the gripper and the held object are known, and it allows users to project this information on the table to allow for higher precision in avoiding obstacles 2b. They tested these two interfaces along with one that did not use cues on a pick-and-place task in a cluttered environment. They found the AR cues to be somewhat helpful with slightly better performance showing that these kinds of cues can be useful. However, they noted that users appeared to be overwhelmed by the amount of information. Meanwhile, advanced cues require 3D models of manipulated objects which will often not be available.



(a) Basic Cues



(b) Advanced Cues

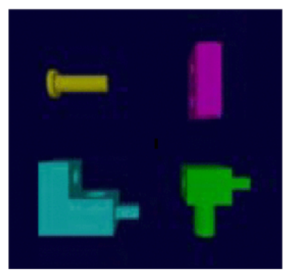
Fig. 2: Difference between interfaces

Lastly, a paper [4] examines the effect of AR interfaces on trust and evaluates interfaces based on 3 influencing factors: robot performance, human safety, and AR information. The paper notes the difficulty of designing AR interfaces and proposes the use of surveys to design interfaces. The paper recommends that AR designs focus on reducing the amount of skill required for operation and improving the situational awareness of the operator.

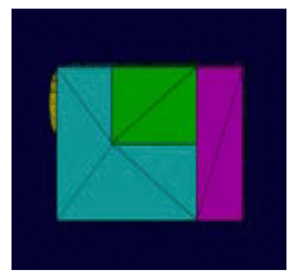
B. Motion Planning and Constraints

In the domain of autonomous robotics, motion planning algorithms have been extensively studied to navigate robots from start to goal positions efficiently. A key challenge is incorporating user-specified constraints, which are essential for safe and effective robot operation in environments shared with humans. Previous work has explored various methods for integrating these constraints into robot planning processes.

For example, a paper [17] sought to solve the problem of unconstrained teleoperated movement via task-aware constraints. The experiment uses a Visual-Reality platform and involves an assembly task where pieces 3a need to be fit together in certain ways 3b. The proposed method uses a system of constraints along with task information to infer the current state of the task and assign the user the most relevant constraint given that information. This paper makes several large contributions which are particularly important for our project, the use of motion constraints to aid in freeform teleop being the largest one. However, it is severely limited by its age as the paper is from 2005 which predates most technological developments in the field of mixed reality.



(a) Individual block pieces



(b) Fitting blocks together

Fig. 3: Impact of constrained motion

Early works in motion planning primarily focused on creating algorithms such as Dijkstra's and A*, which effectively compute the shortest path from start to goal in a known environment. However, these algorithms often fail to incorporate complex constraints that may arise from user requirements or environmental conditions. Significant progress has been made in the development of algorithms that can consider constraints imposed by the environment or task requirements. These include, but are not limited to, avoiding obstacles, adhering to speed limits, and respecting one-way paths.

One notable methodology involves presenting alternative paths to users, who then rank these paths according to their preferences [5]. This ranking process iteratively refines the algorithm's understanding of user priorities, ultimately converging on a user-optimal path that balances the importance of various constraints. Practical implementations of this approach have demonstrated its effectiveness in industrial material transport scenarios, as evidenced by significant reductions in task completion times while accommodating user-defined constraints.

C. Teleoperation and Shared Autonomy

Teleoperation systems and shared autonomy frameworks have shown promise in bridging the gap between autonomous robots and human operators, facilitating complex manipulation tasks. Teleoperated robots, while effective, demand constant human attention, whereas fully autonomous robots struggle in unstructured and dynamic environments.

Recently, simple graphical user interfaces (GUI) and point-and-click paradigms have been developed to further enhance the operator's experience [6]. These interfaces display real-time video augmented with relevant information about the robot's state, allowing for intuitive interaction and efficient task execution.

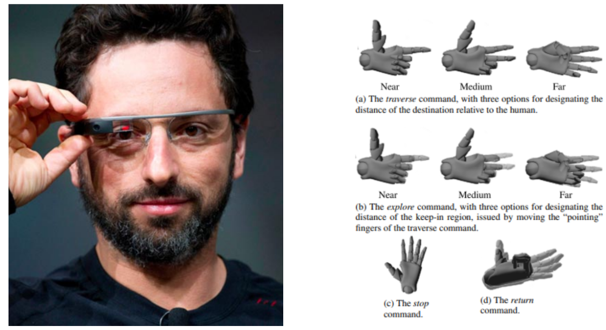
The integration of teleoperation and autonomous capabilities within a single interface allows for seamless transitions between different levels of control, adapting to the task's complexity and the operator's expertise. This hybrid approach not only improves task performance but also ensures a more accessible and user-friendly human-robot interaction.

Lastly, a paper [1] from WPI HiRo lab sought to address the challenges of teleoperation via motion tracking interfaces. For example, when trying to translate the end effector via motion tracking, it is common for users to add accidental rotation to their input. This paper proposes separating out degrees of freedom, for example, using one motion-tracking controller purely for translation and either a set of buttons or another controller for purely rotation. This approach is evaluated via dexterous motion tasks such as reaching, grabbing, moving, and placing. There are many valuable contributions made by this paper, especially the fact that users do not like to manually switch constraint modes during a task. One large limitation of this methodology is the lack of feedback, as it is performed without an AR headset to provide visual cues.

D. Robot Manipulation and Gesture Control

The field of robot manipulation has seen significant progress, particularly in insertion and constrained motion tasks. Soft growing robots, notable for their safe interaction and ease of production, offer advantages in delicate interactions and adaptation to unknown environments. Traditional control methods for robot manipulation have used various physical devices, but recent research focuses on more intuitive interfaces [7]. The "Body Interface" allows human operators to control soft growing robots using arm movements, providing a natural and easy teleoperation experience. Experimental results demonstrate the effectiveness of this interface, with users successfully completing manipulation tasks and achieving high accuracy. These advancements emphasize the importance of intuitive control methods in teleoperation scenarios, reducing the cognitive load on operators and enhancing task efficiency.

Another 2017 paper [16] addresses the problem of a natural way to control a humanoid robot using head gestures, which are detected by Google Glasses. Traditional hand-based inputs are very limited for complex tasks, so their methodology utilized Google Glass sensors, which used accelerometer, gyroscope, and geomagnetic sensors with a complementary filter to reduce noise and drift, such that these head gestures could be detected and be sent to the robot for action as shown in Figure 4. They tested their system on an obstacle avoidance task which demonstrated hands-free natural interaction. They discovered that potential sensor noise drift affected the robustness of gesture recognition.



(a) Google glasses (b) Hand gestures

Fig. 4: Hand gestures on google glasses

III. Methodology

We create two different interfaces to complete an insertion task and a surface wiping task. The first interface is basic AR control of the end effector while the second interface tries to implement features to improve task efficiency for each task.

A. Basic Interface

From a previous WPI project [16], we adopted an AR interface that uses Microsoft HoloLens 2 to control the robotic arm (end effector) of a stationary LocoBot. The existing interface enables users to move the end effector freely in any direction, including up, down, left, right, and perform rotations to grasp objects of varying sizes. Additionally, the interface supported scanning QR codes and AprilTags to precisely locate objects within the physical environment.

B. Proposed Interface

For the insertion task we propose an AR interface which modifies the basic interface with the addition of different constraint modes. These modes constrain motion with respect to a defined surface and a polygon on that surface. The user is able to toggle between the modes using speech commands. Each mode constrains the motion of the end effector so it can only move in 1 or 2 degrees of freedom instead of 6.

The first constraint mode is glide mode. In this mode, end effector motion is constrained to be parallel to the surface-of-interest. Additionally, our polygon-of-interest defines the bounds within which the end effector must stay. In glide mode we show the AR cue seen in Figure 6b to tell the user that it is active. For an insertion task, this mode would allow the user to position the rod that is to be inserted above/in-front of the hole it is to be inserted into.

The second constraint mode is rotate mode. This mode constrains end effector motion to rotation in the axis perpendicular to the surface-of-interest. The AR cue shown for this mode can be seen in Figure 6c. For the example task of using a key to unlock a lock, this mode would allow the user to rotate the key in the lock.

The third constraint mode is insert mode. In this mode motion is constrained to the axis perpendicular to the surface-of-interest. The AR cue shown for this mode can be seen in Figure 6a. For an insertion task, this mode would allow the user to insert the rod into the hole it is meant to go in.

Finally, we implement the command ”rectify” which allows the user to align the orientation of the end effector to be perpendicular to surface of interest. In an insertion task, this would allow the user to pick up rods in one orientation and insert them in a hole in a different orientation.

We hypothesize that these motion constraints will improve the quality of the insertion tasks, both in terms of human workload, as well as task performance.

Because our motion constrained interface requires predefined alignment polygons, i.e. a polygon in 3 dimensional space along which to constrain motion, we propose an accompanying interface for allowing users to define such a surface. The polygon creation interface provides the user with a continuous supply of small spheres hovering above their left palm, using their right hand the user is able to pinch and drag the point into the workspace in 3 dimensions. These points are then normalized into a common plane, and the polygon they create is triangulated, creating a surface mesh. This mesh can then be used for constrained motion control, autonomous wiping, collision checks, etc.

IV. Experiment

Our experiment consisted of a multi-trial user study (n=8) which consists of both objective and subjective metrics to test the efficacy of our proposed methodology.

A. Task Setup

The first task is to insert square pegs into a box. The task setup can be seen in Figure 5a. We have a box with 4 holes for pegs on the top and we have the 4 corresponding pegs that need to be inserted into the box. Pegs come in two different sizes and two different colors. The box has two holes that are aligned with the sides of the box and two holes that are rotated $45°$. The holes were made to have a side length that is 1cm greater than the side length of the corresponding peg allowing for a decent amount of play.

Pegs were placed one-by-one, vertically, in a set position next to the box and users had the task to insert the peg into the corresponding hole in the box. Users have the option to ask for a reset if they get stuck in an unrecoverable state. For each peg we measured completion time, number of collisions, and number of resets.

The second task involves using the hololens and robot arm to “wipe” a space in 3D space in the shape of a chosen polygon in a specific orientation. An example of this task is shown in Figure 7a. The user is given AR queues in the form of dots distributed atop the surface to be wiped.

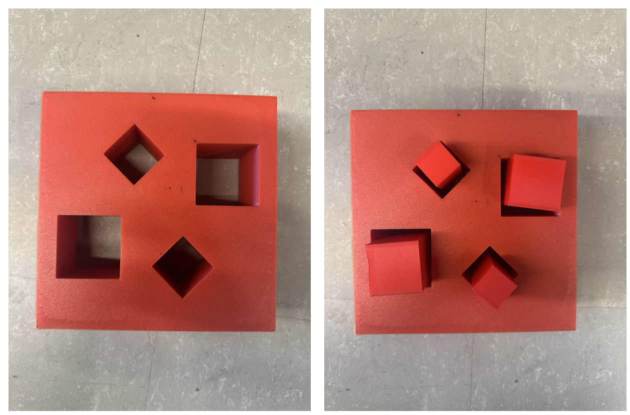
In manual control, the surface is wiped by moving the end effector around the surface. When the end effector comes close to a dot it is marked as wiped and it changes color. Manual trials are considered wiped once all dots have changed color.

The task differs slightly in autonomous mode. Using the polygon creation interface the user defines a polygon to be wiped, this polygon is considered adequate when it intersects all dots. Accuracy can also be determined based on the difference between the angle of the polygon’s face and the angle of the wiping surface. For the scope of this project the automatic task is considered finished when the polygon is defined, there is no actual executed autonomous trajectory. This decision was made in order to prioritize evaluating the interface rather than the quality of motion planning.

B. Procedure

The user study is divided into three parts: filling out a demographic survey, completing either the Insertion or Wipe task, and finishing a post-task survey, including the NASA-TLX survey.

Before operating the robot, users fill out a survey with information including age, sex, occupation, experience with controller based hobbies, experience with AR/VR, and if they are prone to motion sickness.

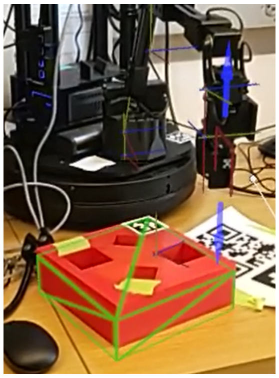


(a) Task Setup (b) Completed Task

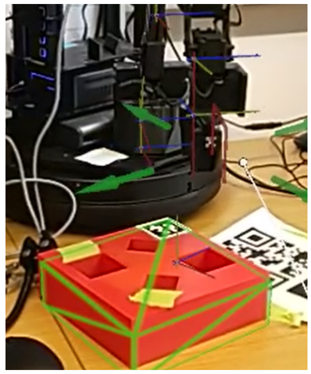
Fig. 5: Insertion Task

The operator is then introduced to the robot and the two insertion interfaces. After 2 minutes of practice time, the user begins trials, starting with a random interface. The insertion task is administered 2 times with each interface. After each trial the user takes a besoke survey to provide some subjective feedback as well as to determine how mentally and physically demanding the interface was. After both trials for an interface, participants take a NASA Task Loading Index (NASA-TLX) survey. Upon completion of the insertion task, an additional bespoke survey is administered asking the user to compare the interfaces. Additionally, during the task, objective factors are measured, including completion time, success rate, and number of collisions will be observed.

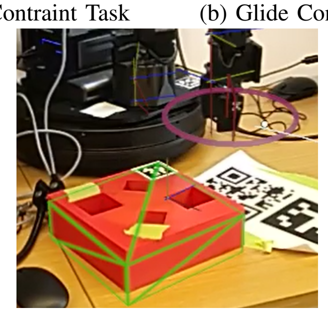
Next the wiping task is administered. First, the participant is given two minutes to get familiar with the polygon definition



(a) Insert Contraint Task

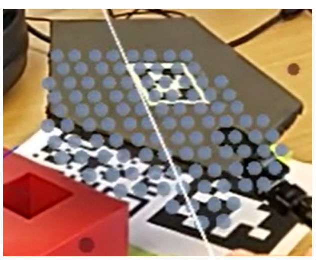


(b) Glide Constraint Task

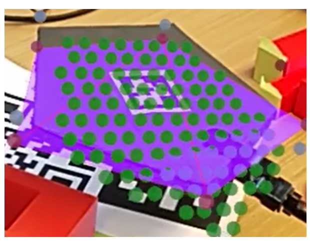


(c) Rotate Constraint Task

Fig. 6: Constrained Tasks



(a) Wiping task



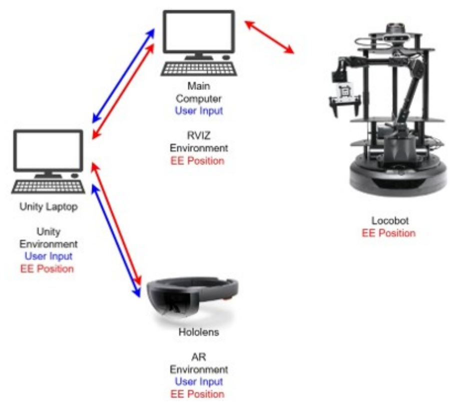
(b) Initial Polygon Definition Interface

Fig. 7: Wipe Tasks

interface. Then they will be given three rounds of polygons to wipe. The complexity of each shape and orientation will increase each round. Each round participants will both complete the task manually and will define the polygon using the polygon definition interface, with the order being chosen randomly. Similarly to the insertion task, participants will be surveyed with a bespoke survey after each task, a NASA-TLX survey after each interface type, and a comparitive survey at the end. For this task, completion time and wiping accuracy will be measured.

V. Platform and Evaluation

The platform design consists of three main components: the Main Computer, the Locobot, and the Unity Laptop. The Main Computer handles the ROS Master, Camera Nodes, User Input and Metric Listener Scripts, and the RViz Environment, ensuring smooth communication and visualization. The Locobot is responsible for executing robotic tasks with its Arm Nodes and Arm Control Scripts. Lastly, the Unity Laptop is utilized for the Unity Environment and the HoloLens Stream, facilitating immersive interaction and control.



(a) Platform overview

Fig. 8: Wipe Tasks

The evaluation process consisted of two primary tasks: the Insert Task and the Wipe Task. For the Insert Task, each user completed two trials using both interfaces, with each trial involving the insertion of one small peg and one large peg at different orientations. Collisions between the peg and the top surface of the box were recorded, and key metrics included the interface type, task completion time, number of collisions, resets, and toggles. Similarly, the Wipe Task involved two trials for each interface, during which bad points (representing out-of-bounds areas) were recorded. Metrics for this task included the interface type, task completion time, and the count of both good and bad points, providing a comprehensive assessment of interface performance and user accuracy.

VI. Results

The following results were collected during a total of 9 user studies. Some users were unable to stay for the extended study meaning there is more data related to the insertion task (n=8) than the wiping task (n=6). While data was collected from a multitude of objective and subjective streams, there is not enough data to preform meaningful statistical analysis. Task performance metrics will mostly be viewed as averages on a by-participant basis to account for skill differences.

The chart in Figure 9 consists of user experience data for two interfaces: Insert Manual and Insert Constraint, across six NASA-TLX metrics: Mental Demand, Physical Demand, Temporal Demand, Performance, Effort, and Frustration.

- Cognitive Load: The Insert Constraint method imposes greater mental and temporal demands. This suggests that the interface was less intuitive and felt like it took a frustratingly long amount of time to use.
- Physical Demand: It appears that the manual interface was significantly more physically demanding. This is likely due to having to manage all 4 Dof at a time instead of just 1 or 2.

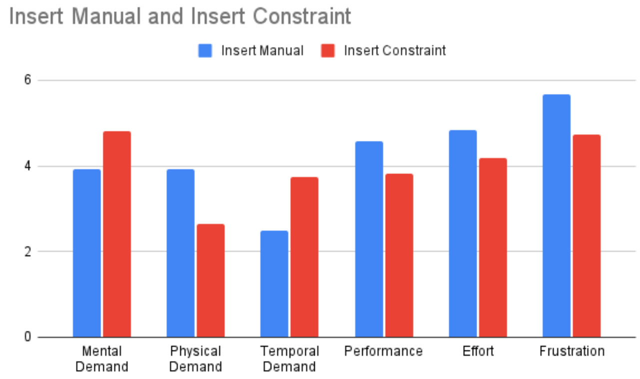


Fig. 9: Self Reported Insert Interface Traits

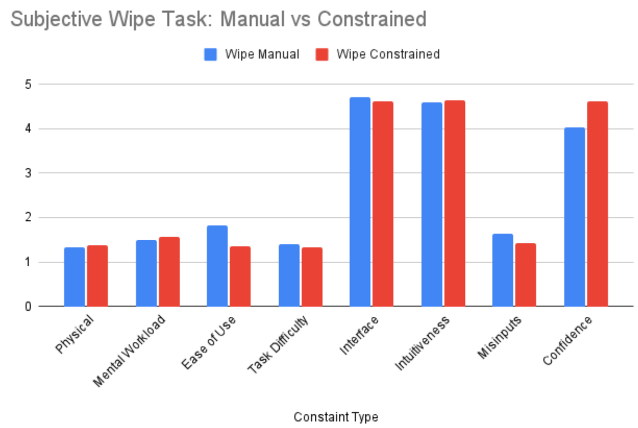


Fig. 10: Self Reported Wipe Interface Traits

- Effort and Frustration: The manual interface showed higher effort and frustration. This means that the constraints allowed users to control the arm in a more manageable way.

The chart in Figure 10 compares subjective evaluations for the Wipe Task between two interaction modes: Wipe Manual (blue) and Wipe Constrained (red), across seven key metrics: Physical, Mental Workload, Ease of Use, Task Difficulty, Interface, Intuitiveness, Misinputs, and Confidence.

- Cognitive Load and Usability: Wipe Constrained places a higher cognitive demand on users (mental workload and misinputs)
- Perceived Accuracy and Precision: Despite its challenges, Wipe Constrained instills greater user confidence, likely due to its structured approach, which users might associate with better precision or reliability.

When viewing the survey data regarding the insert interfaces in Figure 9, as well as the wipe interfaces in Figure 10, it is clear that while both constrained and unconstrained interfaces preform similarly in many metrics, the manual interfaces under preform in confidence compared to their assisted counterparts. Users consistently reported feeling more confident

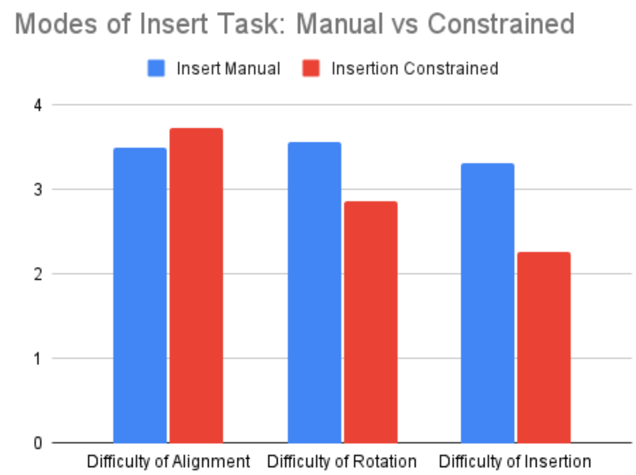


Fig. 11: Self-Reported Constraints Modes Traits

inputs would cause the desired outputs while using the motion constrained insertion, as well as the polygon drawing tool.

The chart in Figure 11 illustrates the user-reported difficulty levels for the three modes of the constrained motion Insert Task—Glide, Rotate, and Insert—compared to the manual task. These modes correspond to the respective tasks of Alignment, Rotation, and Insertion. The Constrained method scored higher in difficulty for alignment tasks, indicating that users found it more challenging to align the peg with the hole compared to manual insertion. Conversely, for rotation and insertion tasks, the constrained method showed lower difficulty ratings than the manual approach. This suggests that the constrained method simplifies and supports users in performing these steps more effectively. Overall, the results highlight the constrained interface's ability to assist with complex manipulations like rotation and insertion, while the alignment process may require further optimization to improve usability.

For insertion tasks, when viewing collisions with unconstrained vs constrained motion (Figure 12) constrained motion almost always preforms better than unconstrained control. For one participant this difference was quite pronounced, with collisions being a virtual non factor for constrained motion, while being a major obstacle in task performance when using unconstrained control. The chart in Figure 12 comparing collisions during the insertion task shows that the manual interface consistently results in more safety hazards than the constrained interface, indicating that the latter enhances safety by reducing collisions. Participants 1 and 6 had particularly high collision counts in the manual mode, suggesting difficulties with precise control. Overall, the constrained interface demonstrated significantly fewer collisions across all participants, highlighting its effectiveness in mitigating errors through guided operation.

The chart in Figure 13 comparing completion times for the insertion task shows that the manual interface generally allows for faster task completion compared to the constrained

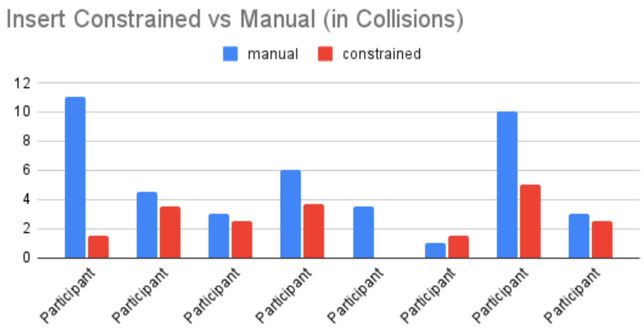


Fig. 12: Collisions with each interface by participant

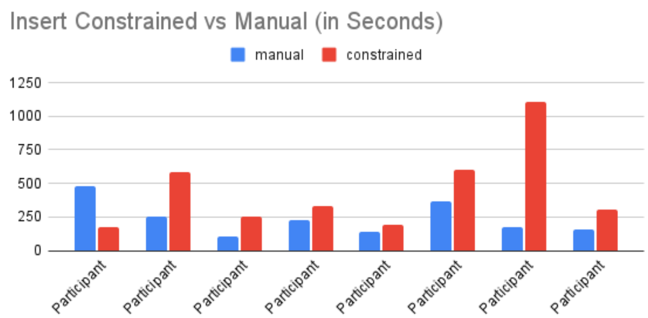


Fig. 13: Task completion time with each interface by participant

interface, though this comes at the cost of safety and precision. Participant 6 notably took much longer to complete the task in the constrained mode, suggesting difficulty adapting to its restrictions. Overall, the constrained interface prioritizes safety but slows down performance, reflecting a trade-off between speed and control.

Interestingly according to Figure 13 while interface choice has a large impact on task time, the preferred interface varies by participant. One possible reason for this is Participant 2's existing experience with robot control providing an advantage with manual control. Overall the manual interface prioritizes speed but leads to more frequent safety hazards (collisions), while the constrained interface improves safety at the cost of longer task completion times. These findings underscore a trade-off between efficiency and safety, suggesting that the ideal interface might balance the two or adapt to individual user needs.

Upon analyzing objective metrics from the Wipe task, the chart in Figure 14 showcases significant variability in user performance across both the manual and constrained interfaces for the wiping task. While some participants, such as Participant 2, demonstrated improved performance with the constrained (polygon definition) interface, others, like Participant 7, performed better using the manual interface. The results indicate that user performance is highly dependent on individual skill, familiarity, or adaptability to the interface,

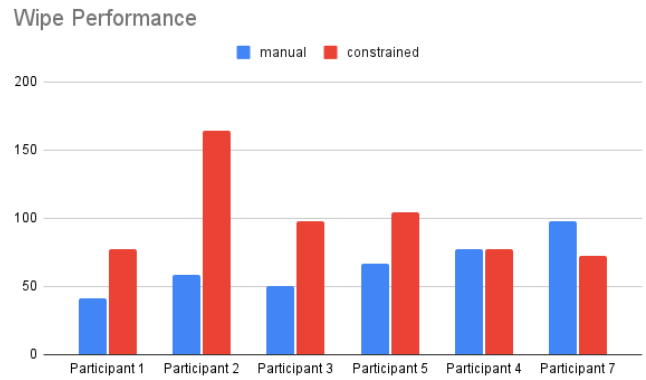


Fig. 14: Wipe performance score with each Wipe interface by participant

emphasizing the need for further exploration of user-specific preferences and potential interface customizations to optimize performance across diverse participants.

VII. Conclusion

A. Advantages

The advantages of the system are evident in both the methods and procedures employed. The methods demonstrate a decreased collision count, increased wipe accuracy, and improved task completion time, particularly for inexperienced users.

B. Limitations

For the proposed insertion interface, one of the main limitations was the use of speech commands to toggle between modes. Since speech commands takes a few seconds to trigger, mode toggles are very costly. This suggests that the efficiency of the interface could be greatly improved with a quicker interface such as a controller with buttons. In addition to increasing speed, this method could also decrease collisions by allowing users to quickly switch to a different mode to make a small adjustment before returning to the current mode.

Another limitation of our proposed insertion interface is that it has only been used with meshes that are in a plane. Future work should extend this interface to work for more general meshes. For general meshes, the end effector should stay oriented such it always points towards the closest point on the mesh. The modes would also have to be changed as follows: glide mode should attempt to maintain a certain shortest distance to the mesh, rotate Mode should rotate the end effector around its own z axis, and insert mode should move the end effector in the direction of its z axis.

For our polygon definition interface, our interface is limited in several ways. First, we are currently using an algorithm that only works for convex polygons and also requires points to be added in a clockwise order. There are existing algorithms that could be implemented that would allow for the use of concave polygons. Additionally, the user interface could be adjusted to allow for adding points anywhere along the perimeter of the surface instead of requiring that new points define the next clockwise vertex of the polygon.

Another limitation of the polygon definition interface is that it only creates meshes that exist in a plane. If we had a task that would involve painting a curved surface, it would be desirable to be able to define a curved mesh. In general, the task of going from a set of points to a mesh is not well defined because their are many possible solutions. This means that a more complex interface would be required. One can imagine a different kind of interface that allows for taking a mesh and stretching and shaping it into the desired shape.

A limitation of the project as a whole is that although the interfaces are designed to work together, we only tested them separately. Future work should use both in combination to see if the interface can be used to improve efficiency for more complex tasks.

Finally our evaluation is limited on sample size. With more user studies and a larger variety of participants we may be able to draw more statistically significant conclusions.

VIII. Task Division

The Unity team (Dimitri, Jaskrit and Aman) works on Hololens code development, including both free and constrained robot arm movements as well as UI elements for the wiping task. The ROS team (Dimitri and Antonio) is responsible for processing commands given from the Hololens and executing arm movements. The entire team collaborates to design and conduct the user study as well as evaluate results. All members also contribute towards writing the paper and preparing the presentation.

References

[1] A. U. Krishnan, T. -C. Lin and Z. Li, "Design Interface Mapping for Efficient Free-form Tele-manipulation," 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Kyoto, Japan, 2022, pp. 6221-6226, doi: 10.1109/IROS47612.2022.9982149. keywords: Tracking;Position control;Virtual reality;Interference;Task analysis;Motion control;Intelligent robots,

[2] K. Chandan, V. Kudalkar, X. Li and S. Zhang, "ARROCH: Augmented Reality for Robots Collaborating with a Human," 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 2021, pp. 3787-3793.

[3] C. Cruz Ulloa, D. Domínguez, J. Del Cerro, and A. Barrientos, "A Mixed-Reality Tele-Operation Method for High-Level Control of a Legged-Manipulator Robot," Sensors, vol. 22, no. 21, p. 8146, Oct. 2022.

[4] R. Palmarini et al., "Designing an AR interface to improve trust in Human-Robots collaboration," Procedia CIRP, vol. 70, pp. 350–355, 2018.

[5] N. Wilde, D. Kulić and S. L. Smith, "Learning User Preferences in Robot Motion Planning Through Interaction," 2018 IEEE International Conference on Robotics and Automation (ICRA), Brisbane, QLD, Australia, 2018, pp. 619-626.

[6] P. Birkenkampf, D. Leidner and C. Borst, "A knowledge-driven shared autonomy human-robot interface for tablet computers," 2014 IEEE-RAS International Conference on Humanoid Robots, Madrid, Spain, 2014, pp. 152-159.

[7] F. Stroppa, M. Luo, K. Yoshida, M. M. Coad, L. H. Blumenschein and A. M. Okamura, "Human Interface for Teleoperated Object Manipulation with a Soft Growing Robot," 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 2020, pp. 726-732.

[8] Stephanie Arevalo Arboleda, Franziska Rücker, T. Dierks, and J. Gerken, "Assisting Manipulation and Grasping in Robot Teleoperation with Augmented Reality Visual Cues," Human Factors in Computing Systems, May 2021, doi: https://doi.org/10.1145/3411764.3445398.

[9] A. U. Krishnan, T. -C. Lin and Z. Li, "Design Interface Mapping for Efficient Free-form Tele-manipulation," 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Kyoto, Japan, 2022, pp. 6221-6226, doi: 10.1109/IROS47612.2022.9982149.

[10] C. P. Quintero, S. Li, M. K. Pan, W. P. Chan, H. F. Machiel Van der Loos and E. Croft, "Robot Programming Through Augmented Trajectories in Augmented Reality," 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, 2018, pp. 1838-1844, doi: 10.1109/IROS.2018.8593700.

[11] Gregory, Jason Reardon, Christopher Lee, Kevin White, Geoffrey Ng, Ki Sims, Caitlyn. (2019). Enabling Intuitive Human-Robot Teaming Using Augmented Reality and Gesture Control. 10.48550/arXiv.1909.06415. G. Quere et al., "Shared Control Templates for Assistive Robotics," 2020 IEEE International Conference on Robotics and Automation (ICRA), Paris, France, 2020, pp. 1956-1962, doi: 10.1109/ICRA40945.2020.9197041.

[12] J. Islam, A. Ghosh, M. I. Iqbal, S. Meem and N. Ahmad, "Integration of Home Assistance with a Gesture Controlled Robotic Arm," 2020 IEEE Region 10 Symposium (TENSYMP), Dhaka, Bangladesh, 2020, pp. 266-270, doi: 10.1109/TENSYMP50017.2020.9230893.

[13] R. Suzuki, A. Karim, T. Xia, H. Hedayati, and N. Marquardt, "Augmented Reality and Robotics: A Survey and Taxonomy for AR-enhanced Human-Robot Interaction and Robotic Interfaces," doi: https://doi.org/10.1145/3491102.3517719.

[14] Y. -H. Su, C. -Y. Chen, S. -L. Cheng, C. -H. Ko and K. -Y. Young, "Development of a 3D AR-Based Interface for Industrial Robot Manipulators," 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Miyazaki, Japan, 2018, pp. 1809-1814, doi: 10.1109/SMC.2018.00313.

[15] Y. Zhong, B. Shirinzadeh, W. Ma and H. C. Liaw, "Assembly Modelling Through Constraint-based Manipulations in A Virtual Reality Environment," TENCON 2005 - 2005 IEEE Region 10 Conference, Melbourne, VIC, Australia, 2005, pp. 1-6, doi: 10.1109/TENCON.2005.301317.

[16] Z. Wang, X. Wen, Y. Song, X. Mao, W. Li and G. Chen, "Navigation of a humanoid robot via head gestures based on global and local live videos on Google Glass," 2017 IEEE International Instrumentation and Measurement Technology Conference (I2MTC), Turin, Italy, 2017, pp. 1-6, doi: 10.1109/I2MTC.2017.7969814.

[17] Y. Zhong, B. Shirinzadeh, W. Ma and H. C. Liaw, "Assembly Modelling Through Constraint-based Manipulations in A Virtual Reality Environment," *TENCON 2005 - 2005 IEEE Region 10 Conference*, Melbourne, VIC, Australia, 2005, pp. 1-6, doi: 10.1109/TENCON.2005.301317.