Augmented Reality-Assisted Reconfiguration and Workspace Visualisation of Malleable Robots

Alex Ranne[†], Angus B. Clark[†], *Student Member, IEEE*, Nicolas Rojas, *Member, IEEE*

Abstract—Malleable robots are a type of reconfigurable serial robot capable of adapting their topology, through the use of variable stiffness malleable links, to desired tasks and workspaces by varying the relative positioning between their revolute joints. However, their reconfiguration is non-trivial, lacking intuitive communication between the human and the robot, and a method of efficiently aligning the end-effector to a desired position. In this paper, we present the design of an interactive augmented reality alignment interface, which helps a malleable robot understand the users task requirements, visualises to the user the requested robot configuration and its workspace, and guides the user in reconfiguring the robot to achieve that configuration. Through motion tracking of a physical two-degree-of-freedom malleable robot, which can achieve an infinite number of workspaces, we compute the accuracy of the system in terms of initial calibration and overall accuracy, and demonstrate its viability. Results demonstrated a good performance with an average repositioning accuracy of 9.64 ± 2.06 mm, and an average base alignment accuracy of 10.54 ± 4.32 mm, in an environment of size 2000mm x 2000mm x 1200mm.

Index Terms—Human-Robot Interaction, Manipulators, Augmented Reality.

I. INTRODUCTION

THE difficulty of designing, trajectory planning, and orientation control of a multi-degree of freedom (DOF) robot are common problems faced by researchers in task oriented robot design [1]. Although these issues have been mitigated as robotics continues evolving, with better computational tools developed to control manipulators with higher accuracy and repeatability implemented [2], one aspect that remains challenging, is the development of solutions where a robot needs to complete tasks in collaboration with a user, instead of operating fully autonomously. Despite the progress made in the last decade with the deployment of collaborative robots [3], and in Human-Robot Interaction (HRI) research [4], under an industrial setting, robots still excel at performing precise, accurate, and repetitive tasks while being kept away from humans, since it is still difficult to visualise their intentions in real time and communicate that to a nearby user [5].

In this paper, we introduce an AR enabled reconfiguration interface for serial malleable robots, thus improving the accuracy of end-effector positioning following a robot reconfiguration. Malleable robots are a type of reconfigurable

Fig. 1. The developed augmented reality-assisted reconfiguration system of a malleable robot. The user is shown interacting with the AR assisted points placement scene overlaid on top of the robot base, with the presence of a transparent configuration space. The lectern to the right prints out the instructions for the task the user needs to conduct.

serial robot developed by Clark and Rojas [6], [7], based around the design of a malleable link—a variable stiffness link between the revolute joints of the robot that allows for their variable relative positioning. This results in a low DOF robot having a significantly increased task versatility, due to the variable workspace of the robot. For extrinsic malleable robots, where their reconfiguration is performed externally by a user directly reshaping the malleable link, one of the key issues that arises is performing this alignment accurately. For example, visualising the new configuration of the robot is difficult for a user, and results in a limited positioning accuracy [7]. The developed AR system allows the user to smoothly generate and place a virtual end effector within the maximum reachable space that the robot can move to (referred to as “reconfiguration space”). Using this desired position and orientation, the system can advise optimal end-effector transformations, compute the workspace for each topology, and ultimately overlay them in front of the user’s eyes. This allows for easier and more accurate alignment of the malleable robot to an optimal position by the user.

The significance of Extended Reality in HRI

Despite HRI being a well-studied field, most researchers agree that there is a need for shared understanding of the robot

[†] Authors contributed equally to this work.

This paper includes a supplementary video available at <https://youtu.be/HgHENruNgel>.

Alex Ranne, Angus B. Clark, and Nicolas Rojas are with the REDS Lab, Dyson School of Design Engineering, Imperial College London, 25 Exhibition Road, London SW7 2DB, UK. (e-mail: {alex.ranne17, a.clark17, n.rojas}@imperial.ac.uk)

workspace in order to enhance collaboration and situational awareness. Although this method of interaction is still novel, a study by Solyman et al. [8] with 2D workspace visualisation has demonstrated the potential of these workflows to accelerate robot assembly tasks. However, the biggest drawback of showing the workspace on a PC is that it requires the user to constantly switch between looking at the display and the robot, which can become tiresome. Furthermore, it can be challenging for the user to map a 2D projection of an image to the 3D world, leading to poor positioning accuracy and increasing operation time. This problem is particularly critical in medical applications, where the lack of information on image depth may compromise patient safety due to incorrect placement of visual markers [9]. Overall, these challenges were commonly encountered in 2D workspace visualisation studies in literature, thus demonstrating the importance of a 3D immersive experience.

One solution to solve these issues is taking the surrounding environment completely out of the problem via the use of virtual reality (VR) enabled teleoperation. Unfortunately, this introduces the problem of mapping a user's reference frame to the robot's reference frame, where an inaccurate bi-directional mapping leads to poor user experience [10]. With that said, simpler solutions to that problem has been found within the "Baxter's Homonculus" system developed by Lipton et al. [11], which leverages on off-the-shelf VR goggles to give the user the same point of view as the robot in action by introducing an intermediate "VR control room" that takes control of the mapping. This decouples the provision of sensory stimuli directly from the robot to the user's VR scene, and allows the user to select which of their movements translates to the robot. While VR technology may provide a user-friendly interface, it does not allow the user and the robot to collaborate harmoniously within the same shared space. When evaluating its accuracy, the Baxter's Homonculus study also presents little insight into the degree of error in the mapping, instead focusing on other practical aspects in manufacturing such as time consumption of assembly and number of grasps.

Following the introduction of augmented reality (AR), image overlay of 3D holograms onto the users vision has been made possible. AR systems have seen applications in a wide range of working scenarios, fulfilling the purposes of motion planning [12], control [13], and visualisation [14]. The main benefits of having an AR assistive system was well summarised by Makhataeva et al.'s review [15]. The studies examined in their work displayed varying levels of accuracy, with some AR enabled teleoperation procedures displaying 60-80 % accuracy when trying to replicate motions with digital cameras [16]. Other systems used in neurosurgical applications achieving mean target errors that are as low as 1.11 ± 0.42 mm [17], by incorporating fiducial markers and using intra-operative re-imaging to adjust for tissue shifting.

Recently, the growth in popularity of the Microsoft HoloLens has attracted developers to enter the head-mounted displays (HMD) AR community. Together with the Mixed Reality Toolkit [18], developers are able to create interactions through methods such as gaze, speech, and gesturing in their AR apps, while preserving the benefits of AR. An

Fig. 2. The previously developed 2-DOF malleable robot to be reconfigured, highlighting the point ($P1 - 5$) representation and interpoint distances ($D13$, $D14$, $D23$, $D24$) that define the malleable link topology.

interesting study conducted by Rosen et al. asked the users to label whether collision occurs for a set of trajectories [12], with some users given no AR support, some given 2D AR visualisation, and the remaining given HMD AR support. The results indicated that the users of HMD yielded better accuracy ratings (75%) compared to 2D AR users (65%). One reason behind this was that having a 3D holographic object mitigates errors caused by the user trying to map a 2D projection of 3D objects onto their vision.

The potential of AR/HMD driven robotic systems do not stop there. In Bolano et al.'s work [19], they used 3D cameras and point clouds to predict robot collision via trajectory and swept volume visualisation. What is interesting in this work was the ability for their system to project dynamic robot workspaces to AR, such that it can enhance user awareness and enable path re-planning. Robot reprogramming was also seen in Kousi et al.'s work [20], where markerless AR-based HRI was used to aid the user to control a mobile robot in a production environment in the case of an unexpected error. Virtual buttons were implemented to initiate robot movements, and the robot can provide real time information of active tasks at the workstations. In summary, AR in HMDs is a promising technology that is worth exploiting for enhancing HRI.

II. AUGMENTED REALITY SYSTEM DESIGN AND DEVELOPMENT

The AR application designed in this paper was for a malleable robot system presented in [7], and preliminary discussed in [6]. It is a 2-DOF reconfigurable robot, formed of a vertical revolute joint at the base, a malleable link connecting this base to a second revolute joint, and a rigid link connecting the second revolute joint to the end-effector. This is an extrinsic malleable robot, and thus we believe that there is potential for an AR system to recommend on the end-effector placement.

We deployed our AR system on the 1st generation of Microsoft HoloLens. AR/MR was selected over VR as it allows for direct manipulation and co-located visualisation of the malleable robot reconfiguration. The HoloLens was selected since it possesses many sensors that capture environmental information and the user's gestures by using its 4 environment

Fig. 3. Involute and Cycloid curves traced by the malleable link in both maximum and minimum lengths, demonstrating how the reconfigurable space of P3/4 (blue) and P5 (green) were theoretically calculated.

understanding cameras and one depth camera. All applications for the HoloLens were developed on the Unity game engine (Version 2018.4.27f1), and the Mixed Reality ToolKit (MRTK, Version 2017.4.3.0). Since the HoloLens behaves similarly to a mobile device, no dedicated PC is needed to run the applications, except for loading the application in the first place. However, the developed system does require an additional workstation to compute then generate the robot workspace, as well as the desired end-effector positions. The software OptiTrack Motive was also run using this workstation to perform the motion tracking. The specifications of the workstation were a Ryzen 9 5900X CPU and 64GB RAM.

The Malleable Robot Reconfiguration Workspace

Given a desired end-effector position and orientation, a set of optimal robot topologies (that is, specific reconfigurations) can be generated that achieve this [7]. With that said, the developed system should allow for efficient reconfiguration, by classifying whether the user demands are valid, demonstrating visually what is achievable, and feeding this information back to the user through an augmented reality enabled interface. Ultimately, this implies robustly defining a reconfiguration workspace, defined as the volume in which all points within this space can be reached by the malleable robot through reconfiguration (not through joint movement) of the robot end-effector when it is not fixed in a certain topology.

For the developed AR application, it was necessary to compute the reconfiguration workspace for the end-effector (P5, as shown in Fig. 2), aiding the user in ensuring that the position of the desired end-effector is within it. However, this is difficult due to the variability in the malleable link (bending, extension, compression, and twist).

To simplify the determination of the reconfiguration workspace, twist of the malleable link was not considered. Using the minimum radius of curvature (r_{min}), the involute and cycloid curves of the malleable link at its maximum (L_{max}) and minimum (L_{min}) lengths were computed, shown in Fig. 3. For the malleable link, with y in the vertical direction and x in the horizontal, the involute curve is defined as:

$$x = r(1 - (\cos(t) + (t - a)\sin(t))) \quad (1)$$

$$y = r(\sin(t) - (t - a)\cos(t)), \quad (2)$$

Fig. 4. Results of the reconfiguration space exploration, shown for the end of the malleable link (P3/4 Experimental) and the robot end effector (P5 Experimental). The theoretical reconfiguration space is overlaid for both, along with the positioning of the robot mounting platform.

and the cycloid curve is defined as:

$$x = r(1 - \cos(t)) \quad (3)$$

$$y = r(t - \sin(t)) + L_{min}, \quad (4)$$

where a is the maximum bending angle of the malleable link, t is the variable curvature of the malleable link, and r is the radius of curvature. Values of $r_{min} = 110$, $L_{max} = 700$, $L_{min} = 550$, and $a = 5$ rad were used.

The total area captured within these curves represents the distal joint at the end of the malleable link (P3/4, shown in Fig. 3) reconfiguration space. Then, using the length of the distal link, it was possible to compute the resulting end-effector reconfiguration workspace based on the reconfiguration workspace of P3/4. By then sweeping this 2D area around the vertical base joint, a 3D theoretical reconfiguration workspace of the end-effector was generated.

To validate this, a reconfiguration space exploration experiment was carried out, where the position of P3/4 and P5 were tracked while the robot was moved throughout its reconfiguration space. These 3D results were then rotated around the vertical base axis, removing the z component, to form a 2D plot for increased readability in comparison against the theoretical reconfiguration space. These results can be seen in Fig. 4. The robot mounting platform has also been added, highlighting why negative y values can not be seen. From these results we can see the theoretical reconfiguration space is a good representation of the actual reconfiguration space of the robot, assuming the theoretical space is adjusted to remove all negative y values. A 3D model of the theoretical reconfiguration workspace of the end-effector, with this adjustment made, was then produced for use in the AR application.

With the generated reconfiguration workspace, several design decisions were made with regards to how it will interact with the other holographic objects. Initially, we considered utilising the reconfiguration workspace purely to provide feedback to the user, but not appearing in the user's vision. Then by constantly checking for collisions between the objects, and changing the colour of the holograms once it collides with the reconfiguration workspace. However, following preliminary user feedback, it was recognised that having a visible workspace can not only speed up the reconfiguration, since the user can easily anticipate which positions and orientations are achievable, and also visualise the range of robot motions a priori to confirm that it will not coincide with objects in the physical space. Therefore, the availability to toggle this workspace on and off was provided in the developed system, and an on-demand feasibility checking system (initiated using a button press) was adopted.

Malleable Robot Optimal Solution Visualisation

For the developed AR application, it was also necessary to guide the user in reconfiguring the robot once an ideal configuration was determined. From the topology computation equations, 4 distances (D_{13} , D_{14} , D_{23} , and D_{24} , shown in Fig. 2) are returned that represent the topology of the robot based on the interpoint distances between P_1 at the robot base origin, P_2 directly above the origin, and P_3 and P_4 which define either side of the distal joint.

Visualising solely interpoint distances is quite difficult. In our previous work on end-effector alignment, a non-AR, tracking based alignment strategies was previously proposed [7]. An OptiTrack (Corvallis, Oregon, USA) system with 6 cameras was setup around the malleable robot. The user was given real-time feedback on the error between the expected position and the current position during an alignment, despite it being extremely difficult for the user to use such errors to correct their movement as there are no feedback regarding which direction this error is pointing towards. One alternative approach could be to use the AR interactive environment to mark the desired P3-5 positions in the user's vision, and physically attaching such points on the robot before performing the alignment. However, such system design will impact comfort and ease of use, since having only the 4 points means the user must deduce the position and orientation of the distal link.

To offer a more intuitive alignment strategy, we propose the following: representing the new robot topology using a mesh model of the distal link, located at the final optimal position defined by the 4 distances, and using it to infer the positions of P3-5 (which is trivial since the relative position of P3-5 does not change with respect to the distal link). To obtain this, first the new locations of P_3 and P_4 were determined using the distances. The mesh model is defined with P_3 at $[0, 0, 0]$, with P_4 directly along the z axis. We can then compute the translation of the mesh model based on the new position of P_3 , and then we can rotate the model around P_3 to align to the new P_4 using a rotation matrix. The rotation matrix is computed using the initial model ($\mathbf{p34}_{model} = [0, 0, 50]$) and new world ($\mathbf{p34}_{world} = P_4 - P_3$) vectors that define the orientation of the mesh model.

Coordinate System and Initial Alignment

To combine the reconfiguration space and the optimal solution visualisation algorithm into a human-robot interaction platform, an appropriate AR interface is now developed to enhance the user experience in manipulating malleable robots. Prior to this, this section will introduce the reference frames attached to each holographic object in the AR space, as such relations lay the foundations of an accurate alignment.

The ultimate goal of understanding the scene coordinate systems is to infer the location of the end-effector with respect to the base. The AR environment consists of four coordinate systems $\{H\}$, $\{E\}$, $\{B\}$ and $\{W\}$, illustrated in Fig. 5, and correspond to the HoloLens, the end-effector, the base and the world origin, respectively. Using its four tracking cameras, the HoloLens is able to infer its own transformation $\{H\}$ with respect to a stationary world origin $\{W\}$. It is also able to resolve spatial relationships between virtual objects.

Before proceeding with the AR application reconfiguration, the virtual and real environments must be aligned at the base $\{B\}$ of the robot. This can be performed automatically, with the addition of error metric feedback, however due to the computational limits of the HoloLens this was performed manually by the user, allowing the HoloLens to focus entirely on performing at its maximum rendering frequency. During the initial alignment phase, the table and robot base holograms are freely movable in terms of position, but with its orientation in x and y fixed. This was done to improve the comfort of the alignment process, but more importantly it limits the alignment error and its effects on the system.

By default, the Unity game engine environment used for the development of this app defines all holographic objects with respect to $\{W\}$, hence mathematically it needs to be transformed into frame $\{B\}$. For example, to find ${}^B_E T$, the transformation from the base to the end-effector, given the relationship between the end-effector ${}^E_W T$ with origin, and the base ${}^B_W T$ with the origin:

$${}^B_E T = {}^E_W T^{-1} {}^B_W T \quad (5)$$

Under Unity, this procedure can be accomplished by defining the local position of virtual objects with respect to the base frame $\{B\}$ using a parent-child relationship.

Augmented Reality Interface

We begin by defining the tools that are necessary to perform a pose reconfiguration in the context of malleable robots. This includes all of the information needed to define the desired end-effector position and orientation, as well as the post-reconfiguration robot workspace.

The virtual world consists of two main hologram modules, the first being the robot base, and the second being the lectern. Each module consists of its own suite of peripheral holograms that are designed to assist the user and provide instructions. The idea behind choosing these two objects as the main HRI medium is simple. For the robot base, since it is an integral part to the robot, it is perfectly logical that it is displayed as a hologram to allow for initial alignment with the real world. During development, the use of the robot base alone without it being coupled with the table was also trialled. However,

Fig. 5. Coordinate systems of the scene. $\{W\}$ is the world origin, $\{B\}$ is the robot base, $\{E\}$ is the end effector, and $\{H\}$ is the HoloLens.

the table was kept since it is visually more convenient to align a larger object, and because the table was later used for robot reconfiguration guidance, once an ideal distal link transformation is generated. On the other hand, the choice of the lectern in place of a "tag along" floating canvas was made, even though the latter was seen more commonly in AR applications, and the user use voice commands to advance to the next task or go back to the previous task. This choice was made since the canvas often gets in front of the way of the user and interferes with their work, while having a fixed canvas located at a comfortable position and orientation is more ergonomic. Both hologram modules were defined with respect to frame $\{B\}$.

Following initial alignment, the user interacts almost exclusively with contents on the lectern and the physical robot, with the exception of placing the desired end-effector hologram, which is grouped under the base module.

The lectern module consists of a lectern hologram with a dynamic instruction panel overlaid on top. The user works through the instructions and provides the system feedback via button presses. The lectern design was chosen as it is ergonomically designed for the user to interact with as they are standing. Its slanted and large surface tilts the panel towards the user, so they can conveniently view its content without straining their neck. In addition, it can be placed next to the working environment and take over the job of a traditional 2D display. On the other hand, scene content holograms and functional classes that are related to the reconfiguration of the robot are grouped under the robot base module. This consists of the table, the robot base, the desired end-effector model, the reconfiguration workspace, and the post-reconfiguration robot workspace unique to its topology.

Following initial alignment of the base and the table (Fig. 8a), the user instructs the robot base to 'anchor' via voice command, fixing the base in place. The anchoring process also places the scene origin as a reference point at the centre of the hologram. The instructions that the user then follows are

Fig. 6. The reconfiguration workflow achieved by the AR-HRI system

Fig. 7. Illustration of determination of the initial alignment error, measured using a corresponding point in the holographic and real world.

inline with the reconfiguration workflow presented in Fig. 6, and can be summarised as follows.

First, the user starts a new configuration (Fig. 8a). Then, the user enters Task 1, a scene consisting of the reconfiguration space computed in Section II. From the lectern shown on Fig. 8b, the user can instantiate the hologram of the distal link, then use gaze and gestures to position and orientate it to a desired configuration.

In conjunction, an end-effector checking algorithm was implemented to confirm whether the holograms lie within the configuration space (Fig 8c and d), and further prevents points that construct a topology which is physically unachievable. On the user's side following a press of the "Check" button, the end-effector hologram will change its colour depending on its current state. A "green" colour indicates that it is within the workspace, and a "red" colour suggests the opposite. Once the hologram turns to green, a "Ready" button is activated, and the user is prompted to progress (Fig. 8d). The user then confirms the coordinates of the distal link points P5 and P6, and the system saves the configuration presented in the scene (Fig. 8e), this records the transformation of all instantiated holograms, closes the app, and sends the information to a workstation. The workstation is responsible for computing the

Fig. 8. An illustration of the workflow of the AR-assisted reconfiguration app, where the top row is the scene at the robot base, and the bottom row is the corresponding instruction panel: (a) Initial alignment of the robot base, (b) the reconfiguration workspace of the robot is shown, (c) a holographic end-effector is generated for the user to place at a desired location, (d) checking whether the end-effector is situated within the reconfiguration space, (e) system displaying the vector of the end-effector, (f) reloading the holograms and the user confirms the desired end-effector location, (g), (h), (i) suggested reconfigurations of the robot and the resulting workspaces they generate are shown, (j) user selects a reconfiguration and proceeds with reconfiguration assistance, (k) animated guidance with holograms aids the user in reconfiguration, (l) and a correctly aligned end-effector post-reconfiguration. This workflow can also be viewed in the linked timestamp of the supplementary video at <https://youtu.be/HgHENruNgeI?t=30>.

optimal topologies of the malleable robot, defined by distance geometry using a sampling of dihedral angles between the triangles formed by the desired point and orientation of the end effector and the fixed robot topology [7], which are used to return mesh models of the achievable robot workspaces that obtains the desired end effector position and orientation, as well as the robot topology transformations, represented by

a mesh model of the distal rigid link and distal joint in the new optimal topology configurations.

Scene saving occurs after the user correctly places their desired end-effector in the reconfiguration space, but before the workstation returns the mesh models of achievable workspaces, along with their end-effector robot topologies. The design decision of using scene saving and loading func-

tions over TCP/IP communication between the workbench and the HoloLens was contemplated during development. The implementation of sockets drastically reduces the frames per second on the HoloLens, and negatively impacts the user experience. However, it ensures a seamless transition between the initial points positioning stage and the reconfiguration stage of the workflow, whereas a traditional saving and loading introduces a break point. Ultimately, we prioritised user experience and speed of reconfiguration over ensuring a seamless transition, since a jittery scene greatly diminishes the immersive experience that an AR system should provide.

Reconfiguration and Alignment

To complete the necessary elements needed to define a reconfiguration, the new robot workspace and end-effector models are loaded into the system (Fig. 8g and Fig. 8h). Note that the workspace defined here differs from the reconfiguration workspace, since it is the volume that the robot can reach given a particular end-effector reconfiguration, given that the malleable link is rigid. In other terms, the robot workspace is a subset of the reconfiguration workspace, defined when the malleable link is rigid and the end effector placed at a unique position and orientation.

The system requests the user for a final confirmation of the configuration scene (Fig. 8i). This is to ensure that the scene has successfully loaded the configuration consisting of the desired end-effector pose, and that a robot workspace which successfully encapsulates the end-effector was generated. If any of these conditions were not met, the user is free to reject the configuration and restart the process. If content with the generated workspace, a mesh of the distal link, positioned in the new robot configuration, is shown in the virtual environment (Fig. 8j). This visualises to the user exactly where to reconfigure and position the malleable robot.

Finally, the user is walked through the reconfiguration process via the lectern's instructions. During this stage, animations and dynamic markers as shown in Fig. 8k were designed to guide the user through the procedures to pressurise and de-pressurise the malleable link. Once the user is happy with their positioning of the distal link with the virtual distal link, they can lock it in place completing the malleable robot reconfiguration (Fig. 8l).

III. PERFORMANCE EVALUATION OF THE AR APPLICATION

Initial alignment accuracy

We first evaluate the accuracy of manually aligning the holographic environment to the real environment origin, located at the malleable robot $P1$ (see Fig 2). 6 Optitrack Flex3 cameras surround the robot, with reflective tracking markers located on the points $P1 - 5$ defining the malleable robot topology [6]. The calibration report of the cameras detailed a mean 3D error for overall projection of 0.426 mm. The holographic environment was then manually aligned to the real environment by a user dragging the robot base hologram onto the real robot base. Once satisfied with the alignment, the holographic environment position was locked.

The initial alignment error is difficult to be measured directly via optical tracking, but can be inferred by using 3 randomly distributed sub-points within the achievable P5 theoretical space, defined by the P5 theoretical space when $Y > 0$, shown in Fig 4, which are visually bounded by the reconfiguration space and mathematically described using equations (1) - (4). Since the relative positions between the points and the robot base is constant across real and virtual space, an offset between the virtual and real points can only result from a misalignment in the robot base. From the virtual space, the location of the holographic points $\vec{P}_{H,i}$ with respect to the virtual origin was provided as a vector $\vec{P}_{H,i} = x_{H,i} \hat{i} + y_{H,i} \hat{j} + z_{H,i} \hat{k}$, where $i = 1, 2, 3$, denoting the three sub-points. An arm with a tracked marker at the end of it was then placed in the real world at each of these virtual points. Then, the coordinates of these points in real space relative to the real origin $\vec{P}_{R,i}$ is denoted as $\vec{P}_{R,i} = x_{R,i} \hat{i} + y_{R,i} \hat{j} + z_{R,i} \hat{k}$. We can compute the difference in Cartesian coordinates between the holographic and real points ($\Delta \vec{P}_i = \Delta x_i \hat{i} + \Delta y_i \hat{j} + \Delta z_i \hat{k} = \vec{P}_{H,i} - \vec{P}_{R,i}$), along with the overall offset (the norm) and the average overall offset across the entire sample:

$$Norm = \sqrt{\sum_i (\vec{P}_{H,i} - \vec{P}_{R,i})^2}, \quad (6)$$

To better illustrate this metric, a labelled diagram including the real and holographic points along with the error is defined graphically in Fig. 7. In this experiment, 5 sets of data were collected, and the results are shown in Table I. Theoretically, the ideal alignment accuracy would show an offset of 0 mm, however due to human and system errors a good accuracy can be defined as <10 mm, as demonstrated by other state-of-the-art AR reconfiguration solutions [21], [22].

Reconfiguration accuracy

We next evaluate the overall task being performed by the developed app: aligning the malleable robot to an optimal reconfiguration. 3 sample optimal reconfigurations based on a desired end-effector position captured through the virtual environment were calculated, using the method presented in [7], which consists of sampling reconfigurations that satisfy the end-effector vector, defined by $P5$ and $P6$ generated through distance geometry. From these optimal reconfigurations, the new position of the distal link was generated, and was displayed in the virtual environment as a holographic distal link. Each configuration was reported in terms of actual interpoint distances ($D13$, $D14$, $D23$, and $D24$ as shown in Fig. 2), which were compared with the desired interpoint distances to compute the positioning difference (Δ). By denoting the actual distances with subscript a and desired distances with subscript d , the reconfiguration error is described as:

$$\Delta D13 = ||D13_d - D13_a|| \quad (7)$$

and so on for $\Delta D14$, $\Delta D23$, and $\Delta D24$.

TABLE I
ALIGNMENT ACCURACY RESULTS, SHOWING THE POSITIONAL DIFFERENCE BETWEEN THE HOLOGRAPHIC AND ACTUAL 3 DESIRED ROBOT END EFFECTOR LOCATIONS. ALL DIMENSIONS ARE SHOWN IN MM.

Sample	Point 1				Point 2				Point 3				Average Norm
	Δ_x	Δ_y	Δ_z	Norm	Δ_x	Δ_y	Δ_z	Norm	Δ_x	Δ_y	Δ_z	Norm	
1	-6.40	-3.44	-7.83	10.67	-2.24	5.95	-9.84	11.71	-1.58	-1.08	-6.35	12.87	11.76
2	-7.43	-13.35	13.99	20.72	0.14	-11.57	5.79	12.94	-6.18	-4.10	12.75	14.75	16.13
3	2.64	-0.85	1.52	3.16	4.77	9.71	-1.27	10.89	-4.17	5.61	3.30	7.73	7.26
4	-12.15	4.00	5.34	13.87	-5.59	8.94	6.91	12.61	-7.79	2.41	6.54	10.45	12.31
5	-1.86	2.00	4.73	5.46	-0.41	-1.64	9.06	9.21	-0.76	-0.75	0.06	1.07	5.25

TABLE II
RECONFIGURATION ACCURACY RESULTS, SHOWING THE POSITIONAL DIFFERENCE Δ AND AVERAGE PERCENTAGE ERROR δ BETWEEN THE IDEAL AND ACHIEVED POSITIONING OF THE DISTAL LINK OF THE MALLEABLE ROBOT, DEFINED BY THE FOUR DISTANCES $D13$, $D14$, $D23$, AND $D24$. ALL DIMENSIONS ARE SHOWN IN MM.

User	Sample 1, Δ					Sample 2, Δ					Sample 3, Δ					User Average
	$D13$	$D14$	$D23$	$D24$	δ	$D13$	$D14$	$D23$	$D24$	δ	$D13$	$D14$	$D23$	$D24$	δ	
#01	20.7	18.7	15.5	8.2	2.8%	2.0	3.0	3.3	5.5	1.8%	2.0	7.2	5.2	11.2	2.5%	8.5
#02	11.2	15.0	8.3	6.4	1.8%	8.2	1.6	6.4	0.3	2.6%	9.3	3.4	5.4	0.0	1.8%	6.3
#03	13.4	18.5	11.7	10.9	2.4%	15.9	16.2	11.5	15.1	8.3%	9.1	6.6	2.4	3.3	2.2%	11.2
#04	11.9	3.0	12.0	2.9	1.3%	19.2	16.9	16.4	16.5	10.2%	5.8	9.2	1.6	6.0	2.4%	10.1
#05	16.1	21.4	14.5	14.0	3.0%	15.4	9.5	12.7	8.6	6.9%	0.1	12.6	4.4	15.7	3.2%	12.1

Percentage errors were also calculated to provide context for the absolute errors relative to the interpoint distances:

$$\delta = \frac{1}{4} \left(\frac{\Delta D13}{D13} + \frac{\Delta D14}{D14} + \frac{\Delta D23}{D23} + \frac{\Delta D24}{D24} \right) \quad (8)$$

It may be noted that the length of the distal link could affect the alignment accuracy, as during a reconfiguration it is aligned to a holographic model of itself, and therefore a longer distal link would produce larger visual misalignments making it easier for a user to achieve a higher alignment accuracy. We therefore evaluate the effect the length of the distal link on the alignment accuracy. An identical reconfiguration was performed using different lengths of distal link (0 - 300 mm in steps of 100 mm), each repeated 5 times to obtain an average. For each of these reconfigurations, the error in interpoint distances ($D13$, $D14$, $D23$, and $D24$) was computed, and the results can be seen in Fig.10.

Experimental setup

5 male participants, aged between 20-26 and experienced in robotics but with no experience regarding manipulating malleable robots, were then invited to align the distal link to the desired location viewed in the HoloLens. This was carried out as a service evaluation, and therefore did not require ethics approval. Each participant was required to familiarise themselves with the HoloLens and its control using the built-in HoloLens introduction app prior to the experiment, where all participants were asked to complete the alignment process within the allocated 5 minutes. During this stage, each user is prompted to carry out the built-in calibration process, where the HoloLens uses their interpupillary distance (IPD) to correct for hologram offsets resulted from wearing the HoloLens differently. At the end of the experiment, once the user is satisfied with their alignment, or if the 5 minute time has elapsed, the real robot configuration was locked and subsequently measured using the aforementioned Optitrack motion tracking cameras. These results for each sample and user are shown in Table II.

Following the reconfiguration experiment, each user was asked to completed the NASA-Task Load Index (NASA-TLX) questionnaire. Developed by the NASA Ames Research Laboratory for subjective evaluation of task workload, the NASA-TLX is a widely used, subjective and multi-dimensional survey that measures the relative contributions to mental workload in terms of mental, physical, temporal task demands, effort, frustration, and perceived performance, in the form of a 100-point visual analog score. A lower score (0 - lowest) for each relative contribution represents a lower task workload, with the exception of performance where a perfect performance is represented by a higher score (100 - highest). The TLX has been tested in thousands of studies that involve experimental tasks, ranging from simulated flights to supervisory control simulations [23]. Since there are a wide range of sub-scales in this questionnaire, it provides diagnostic information about the sources of load. The original version of the NASA-TLX test includes an individual weighting process of the listed sub-scales, by letting the subjects compare them pairwise and decide on their importance. In our evaluation, this step was replaced by an average score across the 6 sub-scales. The results of the NASA-TLX questionnaire are shown in Fig. 9.

IV. RESULTS

Results of the alignment accuracy test across 5 sets of samples are shown in Table I. We found the maximum absolute error in individual Cartesian components to be 13.99 mm, and the maximum norm error to be 20.72 mm. On average, all of the norm errors were within 20 mm, and Sample 5 exhibited the lowest alignment error at only 5.25 mm, whereas Sample 2 had the highest average norm error at 16.13 mm. Despite these differences, there were no significant deviations between the average norm errors, and no distinct outliers were identified.

Results of the reconfiguration accuracy test are shown in Table II. Of the three sample configurations all participants achieved a successful reconfiguration, with positional offsets of <21.4 mm, <19.2 mm, and <15.7 mm respectively. The

Fig. 9. NASA-TLX results

Fig. 10. Results on the effect of different distal link lengths on the reconfiguration accuracy, represented by the error in interpoint distances D13, D14, D23, and D24 of the alignment.

respective average percentage errors were similar for sample 1 and 3 at $<3.0\%$ and $<3.2\%$ respectively, while sample 3 showed a higher average percentage error of $<10.2\%$. User #002 produced the lowest average offset across all samples at 6.3 mm, whereas user #005 produced the highest at 12.1 mm. During experimentation, the rendering frequency of the application ranges between 10-30 fps, with the rendering quality kept at the lowest setting.

Finally, the NASA-TLX results can be seen in Fig. 9. Generally, a high (>60) performance score was given, with users believing they performed the task well. A low (<40) frustration, physical workload, and temporal workload score were given, while mental workload and effort had a medium ($40 < s < 60$) score.

V. DISCUSSION

The average alignment error is $10.54 \pm 4.32\text{mm}$. When compared to the manual alignment of the malleable robot previously presented [7], where an error of 19.62mm was shown, this is a significant improvement. Not only that, but in the manual reconfiguration this accuracy was achieved following training on the robot reconfiguration, whereas only limited training on the HoloLens functionality was provided here. When compared to other HMD based systems found in literature, the results are generally inline with expectations. For example, a similar experiment carried out by Fotouhi et

al [21] on virtual-to-real hologram alignment provides a good benchmark for our study. Within their experiment, they found a mean alignment error of $16.5 \pm 11\text{mm}$ after conducting 10 experiments across 4 users. Both in our case and in Fotouhi et al's study, the alignment errors are unneglectable, requiring further improvement to the system to successfully perform tasks under a work setting that demands high accuracy and precision. However, for extrinsic malleable robots aimed at human-robot collaboration tasks where the required accuracy of the task is lower [6], [24], the results presented provide an acceptable accuracy, especially considering the human-aspect of alignment on a non-gravity compensated robot. Given the contribution of this work in malleable robot 3D workspace visualisation, generation and task guidance, and the demonstrated improvement in alignment error compared to its manual counterpart, we therefore conclude that the system is effective in supplementing human robot interactive tasks.

The main sources of alignment error can be summarised as follows. First, the alignment was carried out manually via user gesture inputs, making the system prone to human errors such as difficulties in depth perception. This was expected, since we have not implemented an automated alignment system using fiducial markers like QR codes. Fiducial markers could allow the initial alignment to be carried out automatically, such as with the markers placed on the robot base, however as mentioned this was not performed to reduce the computational complexity of the application. Second, there is an inherent tracking error with the HoloLens when defining the location of the holographic way points and in tracking the user's position due to the sensor's accuracy limitations. Future work should investigate such errors through motion tracking, and implement an automated alignment system. Finally, performance bottlenecks of the HoloLens due to its hardware limitations such as having only 2GB of RAM, have led to low rendering frequencies when handling 3D rendering of large workspaces. According to the HoloLens official documentation [25], hologram stability in such cases is compromised, where swift head movements by the user within the same environment may lead to the system incorrectly assuming that user has changed rooms, and thus lead to unexpected shift in hologram positions. With the provision of updated and better hardware, such issues will be mitigated in the future.

In the perspective of reconfiguration accuracy, the system displayed good outcomes, where the average error across all users is $9.64 \pm 2.06\text{mm}$, which in absolute terms is similar to what was found for the average alignment error. The percentage error in each sample across different users was also found. The average percentage error is $3.54 \pm 2.57\%$, making it better than existing HMD based systems highlighted previously [16]. The main source of error for reconfiguration stems from the malleable robot itself. As the malleable link becomes pressurised, the manipulator stiffens. In certain configurations where the weight of the distal link is acting perpendicular to the malleable link, it is not able to sustain its shape as a rigid body would. Therefore, this may cause a shift in its actual position with respect to the desired position.

Comparing the effect of the distal link length on the reconfiguration accuracy, we can see there is a very slight

trend in decrease in error with shorter distal links. However, this trend is very minimal, with the average error decreasing from 3.92 mm at 300 mm to 3.21 mm at 0 mm. For the distal link length of 200 mm, we even see a slight increase in error. The change in error across the distal link length is within the variation observed across repeated samples due to human error in alignment, therefore we can reasonably say that the distal link length has little effect on the reconfiguration accuracy.

To analyse the user's experience of the AR app, the participants were requested to complete the NASA TLX questionnaire, and its results are presented in Fig. 9, where the solid coloured bars indicate the mean score and the error bars indicate the standard deviation. The participants rated their experience in the context of each sub-scale with a score out of 100. In general, we see that the user experience has been positive. In particular, the low frustration and mental workload scores indicates that the majority of the participants were satisfied with the system experience. Contrastingly, the high mean effort score of nearly 60 highlights that there are still aspects in the overall system that needs improvement. It is suspected that our score was higher due to the effort needed to perform the initial alignment of the base to the correct location, and to reconfigure the end effector to its optimal pose. When comparing this score to AR assistance apps, an average score that is as low as 35 [12] or as high as 60 [26] was found for AR applications in industrial robots, whilst our system's average NASA TLX score was found to be 42.83 ± 15.35 . In the future, this score can be improved by introducing additional system-to-user feedback, such as via haptic control or sound.

VI. CONCLUSIONS

In this paper, we presented the development of an AR-assisted reconfiguration and visualisation application for a malleable robot. The malleable robot is unique since it has the ability to modify its workspace depending on the topology of its malleable link. During development, we simulated the reach of the manipulator in the form of the reconfiguration workspace using involute and cycloid functions, which was verified empirically through an optical tracking experiment. Using distance geometry, we were able to locate the optimal malleable robot distal link solution, then visualise it in the form of a mesh model by transforming the distal link to its desired location. Based on these mesh models and post-reconfiguration robot workspace, an AR user interface was developed to utilise these components to achieve a successful manual robot reconfiguration.

We evaluated the performance of the system in the perspective of the base alignment accuracy and the distal link reconfiguration accuracy. The AR app showed a good performance in base alignment (10.54 ± 4.32 mm) for the case of markerless virtual-to-real registration. For the reconfiguration accuracy, the app exhibited an error of 9.64 ± 2.06 mm, which is also satisfactory provided the stiffness performance limitations of the malleable robot at some configurations. An evaluation of the user interface was provided in the form of a NASA TLX questionnaire, where the perceived workload during a task was measured. The average workload score across the 6 sub-scales

and all the users is 42.83 ± 15.35 , where the lowest scores were found under the frustration criteria. From this score, we conclude that our system does not impose significant overall workload on the user. Finally, errors for alignment and reconfiguring the distal link can be mitigated in future work by using a fiducial marker based alignment system, and by implementing refinements in the malleable robot mechanism.

REFERENCES

- [1] H. Al-Dois, A. Jha, and R. Mishra, "Task-based design optimization of serial robot manipulators," *Engineering Optimization*, vol. 45, no. 6, pp. 647–658, 2013.
- [2] A. Nagabandi, K. Konolige, S. Levine, and V. Kumar, "Deep dynamics models for learning dexterous manipulation," in *Conference on Robot Learning*. PMLR, 2020, pp. 1101–1112.
- [3] F. Sherwani, M. M. Asad, and B. Ibrahim, "Collaborative robots and industrial revolution 4.0 (ir 4.0)," in *2020 International Conference on Emerging Trends in Smart Technologies (ICETST)*. IEEE, 2020, pp. 1–5.
- [4] R. Galin and R. Meshcheryakov, "Review on human–robot interaction during collaboration in a shared workspace," in *International Conference on Interactive Collaborative Robotics*. Springer, 2019, pp. 63–74.
- [5] Z. Zhou, R. Xiong, Y. Wang, and J. Zhang, "Advanced robot programming: A review," *Current Robotics Reports*, pp. 1–8, 2020.
- [6] A. B. Clark and N. Rojas, "Design and workspace characterisation of malleable robots," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 9021–9027.
- [7] A. B. Clark and N. Rojas, "Malleable robots: Reconfigurable robotic arms with continuum links of variable stiffness," *IEEE Transaction on Robotics*, vol. Submitted 11/06/21, 2021.
- [8] A. E. Solyman, K. M. Ibrahim, M. R. Atia, H. I. Saleh, and M. R. Roman, "Perceptive augmented reality-based interface for robot task planning and visualization," *Int J Innov Comput Inf Control*, vol. 16, no. 5, pp. 1769–1785, 2020.
- [9] L. Ma, Z. Zhao, F. Chen, B. Zhang, L. Fu, and H. Liao, "Augmented reality surgical navigation with ultrasound-assisted registration for pedicle screw placement: a pilot study," *International journal of computer assisted radiology and surgery*, vol. 12, no. 12, pp. 2205–2215, 2017.
- [10] M. R. Mine, F. P. Brooks Jr, and C. H. Sequin, "Moving objects in space: exploiting proprioception in virtual-environment interaction," in *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, 1997, pp. 19–26.
- [11] J. I. Lipton, A. J. Fay, and D. Rus, "Baxter's homunculus: Virtual reality spaces for teleoperation in manufacturing," *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 179–186, 2017.
- [12] E. Rosen, D. Whitney, E. Phillips, G. Chien, J. Tompkin, G. Konidaris, and S. Tellex, "Communicating and controlling robot arm motion intent through mixed-reality head-mounted displays," *The International Journal of Robotics Research*, vol. 38, no. 12-13, pp. 1513–1526, 2019.
- [13] M. Ostanin, R. Yagfarov, and A. Klimchik, "Interactive robots control using mixed reality," *IFAC-PapersOnLine*, vol. 52, no. 13, pp. 695–700, 2019.
- [14] J. A. Frank, M. Moorhead, and V. Kapila, "Mobile mixed-reality interfaces that enhance human–robot interaction in shared spaces," *Frontiers in Robotics and AI*, vol. 4, p. 20, 2017.
- [15] Z. Makhataeva and H. A. Varol, "Augmented reality for robotics: a review," *Robotics*, vol. 9, no. 2, p. 21, 2020.
- [16] H. Hedayati, M. Walker, and D. Szafir, "Improving collocated robot teleoperation with augmented reality," in *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, 2018, pp. 78–86.
- [17] D. Guha, N. M. Alotaibi, N. Nguyen, S. Gupta, C. McFaul, and V. X. Yang, "Augmented reality in neurosurgery: a review of current concepts and emerging applications," *Canadian Journal of Neurological Sciences*, vol. 44, no. 3, pp. 235–245, 2017.
- [18] Microsoft, "Hololens 2—overview, features, and specs — microsoft hololens," 2016. [Online]. Available: <https://www.microsoft.com/en-us/hololens/hardware>
- [19] G. Bolano, A. Roennau, and R. Dillmann, "Transparent robot behavior by adding intuitive visual and acoustic feedback to motion replanning," in *2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2018, pp. 1075–1080.

- [20] N. Kousi, C. Stoubos, C. Gkourmelos, G. Michalos, and S. Makris, "Enabling human robot interaction in flexible robotic assembly lines: An augmented reality based software suite," *Procedia CIRP*, vol. 81, pp. 1429–1434, 2019.
- [21] J. Fotouhi, T. Song, A. Mehrfard, G. Taylor, Q. Wang, F. Xian, A. Martin-Gomez, B. Fuerst, M. Armand, M. Unberath *et al.*, "Reflective-ar display: An interaction methodology for virtual-to-real alignment in medical robotics," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2722–2729, 2020.
- [22] F. X. Creighton, M. Unberath, T. Song, Z. Zhao, M. Armand, and J. Carey, "Early feasibility studies of augmented reality navigation for lateral skull base surgery," *Otology & Neurotology*, vol. 41, no. 7, pp. 883–888, 2020.
- [23] S. G. Hart, "Nasa task load index (tlx)," *Language*, vol. 1986, 1986.
- [24] A. B. Clark and N. Rojas, "Stiffness-tuneable limb segment with flexible spine for malleable robots," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 3969–3975.
- [25] A. Turner, H. Ferrone, D. Coulter, and S. Majumdar, "Hologram Stability," 2020. [Online]. Available: <https://docs.microsoft.com/en-us/windows/mixed-reality/develop/platform-capabilities-and-apis/hologram-stability>
- [26] S. Stadler, K. Kain, M. Giuliani, N. Mirnig, G. Stollnberger, and M. Tscheligi, "Augmented reality for industrial robot programmers: Workload analysis for task-based, augmented reality-supported robot control," in *2016 25th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2016, pp. 179–184.