AUGMENTED REALITY BASED DIGITAL TWIN TO CONTROL MRI COMPATIBLE ROBOT

L. (Lennard) Marx

MSC ASSIGNMENT

Committee:
prof. dr. ir. S. Stramigioli
dr. ir. K. Niu
dr. V. Groenhuis, MSc
dr. ir. W.M. Brink

January, 2024

002Ram2024
Robotics and Mechatronics
EEMCS
University of Twente
P.O. Box 217
7500 AE Enschede
The Netherlands

An Augmented-Reality based control method for an MR safe robot in breast biopsy

Lennard Marx[1], Vincent Groenhuis[1], Wyger Brink[1], Stefano Stramigioli[1] and Kenan Niu[1]

***Abstract*— Magnetic Resonance Imaging (MRI)-guided breast biopsies, integral to breast cancer diagnosis, require multiple preparatory and control scans to navigate the biopsy needle to the target and are susceptible to human errors. Robotic interventions offer the potential to automate these procedures while enhancing targeting accuracy. The imperative for practical and intuitive interfaces to keep operators in the loop becomes apparent. Emerging technologies like augmented reality (AR), and head-mounted displays (HMD) present an immersive solution for human-robot interaction and teleoperation. This study introduces an AR-based control method for an MR-safe robot to conduct breast biopsies. Leveraging hand gestures, voice commands, and intuitive interfaces, operators can effectively engage with the robot to perform the biopsy. Features such as auto-targeting and path planning offer crucial support, ensuring a high degree of accuracy and success rates. Experiments within an MRI environment show the efficacy of these features, resulting in an achieved accuracy of 1.36 $\pm$ 0.89 mm. A substantial reduction of time cost of MRI-guided biopsy was observed as one full biopsy procedure from acquisition to final control scan took 11.83 $\pm$ 2.10 min, compared to the 35-39 min in clinical practice. In conclusion, the proposed method demonstrates a promising approach to AR-guided robotic control in breast biopsy. It also provides great potential to be extended to other applications of robotic teleoperation.**

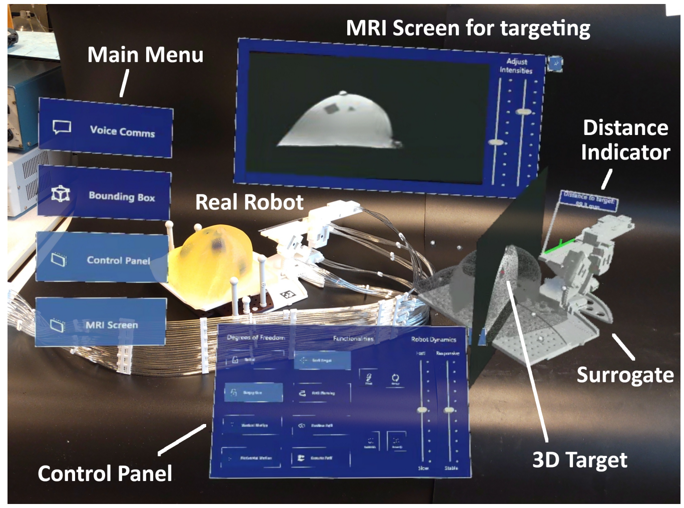


Fig. 1: The point of view of the operator overviewing the holographic scene. To the left is the real robot, which is placed in the MRI, to the right the virtual surrogate. The main menu is fixed in the field of view and is used to toggle the MRI screen, control panel, voice command legend, and the bounding box used to move and scale the scene.

I. INTRODUCTION

Clinical Background

With a mortality rate of 6.9%, breast cancer continues to exert a significant societal impact, contributing to approximately 685,000 annual deaths globally [1], [2]. Representing 31% of female cancers, the likelihood of a woman being diagnosed with breast cancer in her lifetime stands at 1 in 8 [3]. Annually, 2.3 million new cases are identified, marking it as the most frequently diagnosed cancer with an incidence rate of 55.9 per 100,000 [1], [2]. The annual increment of its incidence rate by 0.5% signifies a growing concern regarding its impact [3]. Notably, in countries of the global south, the incidence rates are lower, accompanied by a mortality rate that is 17% higher [1]. This underscores the critical role of the accessibility of enhanced diagnostics in averting breast cancer-related fatalities. Moreover, it suggests that as diagnostics improve, the incidence rates in these regions are anticipated to rise significantly, potentially leading to earlier treatment and a subsequent reduction in mortality rates.

[1]Robotics and Mechatronics, University of Twente, Enschede, AE, The Netherlands `l.marx@student.utwente.nl` (L.M.), `v.groenhuis@utwente.nl` (V.G.), `w.m.brink@utwente.nl` (W.B.), `s.stramigioli@utwente.nl` (S.S.), `k.niu@utwente.nl` (K.N.)

Mammography and ultrasound imaging presently stand as the most commonly employed tools for breast cancer screening and diagnosis. However, their sensitivity in detecting early-stage cancers is relatively low, at 33% and 40%, respectively, while further diminishing with the density of the breast, which is mostly higher in younger women [4], [5], [6], [7]. Even when utilized in combination, their collective sensitivity only marginally improves to 49%, underscoring their limited efficacy in early breast cancer diagnosis. In contrast, magnetic resonance imaging (MRI) shows a superior sensitivity of 91% [4]. Consequently, routine MRI screening is recommended for individuals with a 20% or higher augmented risk of breast cancer, such as those with a familial history of breast or ovarian cancer, or individuals treated for Hodgkin's disease [5]. The primary hindrance to more widespread MRI screening for individuals at lower risk is the associated MRI occupation time and cost [5], [8]. The reduction of the MRI occupation time and cost could potentially facilitate increased screenings, leading to the early detection of more lesions [9]. Upon the identification of a potential lesion, a biopsy to confirm its benign nature has to be conducted.

The current breast biopsy procedures are lengthy, requiring multiple MRI scans to attain and ensure the desired outcome

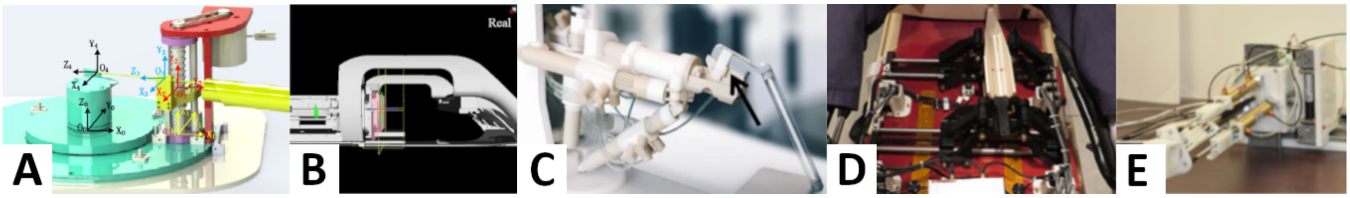


Fig. 2: Various MRI safe robots used for breast and prostate biopsy. A - Song et al. and Liu et al. [11], [12], B - Chan et al. [13], C - Vilanova et al. [14], D - Moreira et al. [15], [16], E - Yang et al. [17]

[10], [8]. Simultaneously, a substantial section of tissue is extracted from the breast to minimize the risk of missing the target, resulting in an extended and stressful period for the patient within the confined space of the MRI bore. Preceding the biopsy, a contrast agent is administered to enhance the visibility of potential lesions. The patient is positioned face-down on the examination table, and the breast is secured in a grid-shaped plastic clamping system [8] simultaneously serving as a reference for the needle insertion. After an initial scan, the patient is removed from the MRI bore, administered local anesthesia, and the biopsy needle, encased in a plastic sheath, is inserted into the breast. The needle is withdrawn, leaving the plastic sheath in place, acting as a placeholder visible on the subsequent control scan. Following a confirmation scan, if necessary, adjustments to the needle needle position are made. Once the needle positioning is deemed satisfactory, the surgeon extracts multiple biopsies around the needle tip, employing a vacuum-assisted needle to retrieve the tissue samples [8]. An additional control scan after the biopsy is conducted, after which additional follow-up biopsies may be performed.

The manual MRI-guided procedure remains susceptible to human error, additionally the use of a flat grid for orientation is restricting the insertion point, which is potentially sub-optimal. Corrections to the needle position may be required, leading to increased tissue damage, patient stress, and longer occupation time of the MRI room. The integration of MR-safe robotic manipulators can assist surgeons in the targeting task, accelerating the process and enhancing accuracy. Various approaches to MR-safe robots are discussed in the current literature. Song et al. as well as Chan et al. developed MR-safe robots dedicated to breast biospies [11], [12], [13]. In both works, the robots are controlled through a graphical user interface (GUI), via mouse and keyboard. Similar robots dedicated to prostate biopsies were developed by Vilnova et al. and Moreira et al. [14], [15], [16]. Both robots act as a guide for the positioning of the needle, the insertion still is conducted by the surgeon. A more intuitive control system was introduced by Yang et al., utilizing an additional controller robot to the peripheral robot in the MRI bore [17]. Through the controller robot, the operator received force-feedback during the needle insertion. While robotic systems are widely recognized as the prospective solution for targeted MRI-guided biopsies and interventions [18], their integration into clinical practice has yet to be implemented. This delay could be attributed to the absence of an intuitive control method, resulting in high training costs for specialized staff, or to an unsubstantial improvement with respect to (w.r.t.) the accuracy and time savings. Augmented reality, with its spatial rendering capabilities, holds a potential starting point to enhance the intuitiveness of human-robot interaction in this context and streamline the biopsy procedure.

AR in Robotic Navigation

The application of Augmented Reality (AR) for robot control varies in its complexity, ranging from basic floating graphical user interfaces (GUI) to intricate gesture and voice-controlled interactive environments. AR can serve as a portable interface, replacing traditional computer screens for robot control purposes (Fig. 3A [19]). These interfaces allow manipulation of the robot's joint space and adjustment of specific control parameters (Fig. 3B, [20]). However, the interactive potential of AR can be fully harnessed through more advanced approaches. Xue Er Shemaine et al. developed a framework utilizing the Robot Operating System (ROS) to interactively control industrial robots. This was achieved by creating floating objects representing waypoints along the intended path for the robot to follow [21]. Similarly, Walker et al. created a virtual surrogate of a drone that can be manipulated and mirrored by the real drone [22]. Expanding on the use of AR, additional communication methods, such as gestures, can be employed for robot control [23]. Cameras in the operator's workspace or integrated into head-mounted displays (HMD) can facilitate this interaction. An advanced example of AR as a robot-human interface is demonstrated in the work of Szczurek et al. [24], where the environment of a remotely controlled robot, as well as the robot itself, is reconstructed in 3D around the operator. Through voice commands and direct interaction with the robot's surrogate, the operator can perform tasks, provide instructions, and define paths, among other functionalities.

The application of Augmented Reality (AR) in surgical settings has been less extensively documented. Lin et al. explored the use of gestures to control a surgical robot in AR, specifically for navigating endoluminal interventions [25]. In this setup, the trachea is represented in the AR space, allowing the operator to define the desired destination of the robot's end effector with hand gestures, which is then followed by the robot. One of the notable potentials of AR in surgical navigation lies in its ability to leverage imaging techniques, particularly from MRI scans. Morales Mojica et al. and Velazco-Garcia et al. demonstrate the use of 3D depictions of MRI images for surgical planning [26], [27].

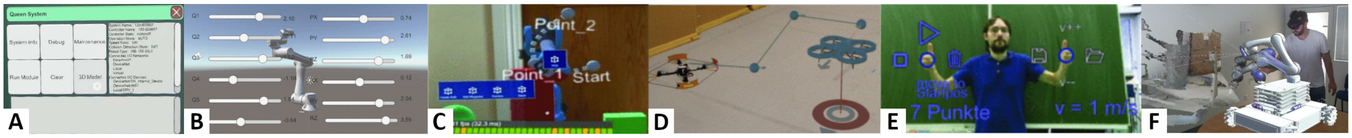
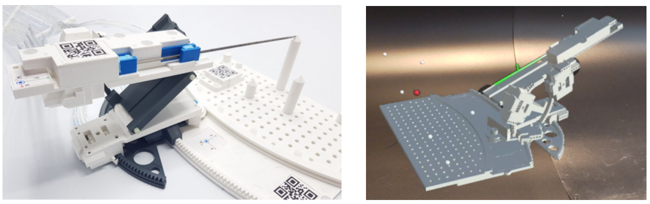


Fig. 3: Various AR based robot control schemes reported in literature. A - Stone et al. [19], B - Islas et al. [20], C - Xue et al. [21], D - Walker et al. [22], E - Thormann et al. [23], F - Szczurek et al. [24]

The 3D visualization of the MRI scan aids in locating the target of the operation and planning the path of the robotic manipulator.



(a) The Sunram7 robot. (b) The virtual surrogate.

Fig. 4: The Sunram7 robot and the virtual surrogate.

This study aims to enhance the efficiency and safety of breast biopsies by introducing an intuitive and accurate teleoperation method for an MR-safe robot. The primary objective is to significantly reduce the time required for the biopsy procedure, consequently lowering costs and potentially extending access to a broader patient population. Such advancements hold the potential to contribute to a reduction in breast cancer mortality rates.

Augmented Reality (AR) serves as a crucial component, providing a visual representation of the scene inside the MRI bore. The operator can import the MRI dataset into the AR application, where it is rendered as a 3D mesh and automatically calibrated to the robot's position. Through the integration of gestures, voice commands, and other features, the robot can be navigated quickly and precisely to target the lesion and perform the biopsy with improved accuracy. The proposed control paradigms are adaptable to various manually or semi-autonomously controlled robotic manipulators.

II. METHODS

System Overview

The MR-safe robot employed in this study is the Sunram7 [28]. To ensure complete MR safety, the robot is constructed entirely from 3D-printed plastic components, except for the MR-conditional titanium needle, and features pneumatic motors that actuate its joints. The robot is designed with five Degrees of Freedom (DOFs), four rotational joints that facilitate movement along both the vertical and horizontal planes, and one translational joint responsible for the back-and-forth movement of the biopsy needle (Fig. 4).

Figure 5 presents an overview of the interactions between the components within the system. The holographic representation of the Sunram7 robot is rendered in AR in front of the operator via the Hololens 2 HMD (Microsoft, Redmond, Washington, U.S.) (Fig. 1). This HMD is controlled through hand gestures, voice commands, and a virtual control panel (Fig. 6). The operator utilizes hand gestures to define the desired position of the end-effector (needle), through inverse kinematics (IK) the required joint angles are calculated (Section II). Subsequently, a controller computes the dynamics with which the robot approaches the desired position (Section II). An MRI scan of the patient's breast - in the case of this study in the form of a phantom - including the robot's base frame is obtained, similarly to traditional biopsy procedures [8]. This dataset is imported, and automatically calibrated relative to the robot's position with a single button click, as detailed in section II. The dataset is then transformed into a 3D mesh, and a 2D plane is positioned within the mesh, enabling the generation of an MRI image corresponding to the plane's position. With the implemented targeting system (Section II), the operator can define a biopsy target in 3D space, to which the robot can automatically align itself. Throughout this automatic targeting process, the operator receives feedback on the distance of the needle tip to the target's center. Upon satisfaction with the position, the biopsy can be taken, and the needle retracted. Alternatively, a path can be planned, previewed, and subsequently executed using the built-in path planning system (Section II).

The cornerstone of the AR-based control scheme is the gesture control, enabling the operator to specify the desired position of the end-effector, by forming a fist with the right hand. This is achieved by manipulating a virtual ray or trajectory, to which the joint space alignment is calculated via the IK. The basis for the positioning of the ray is the operator's hand when forming the according gesture. Since the position of the hand is estimated by the outside-facing cameras of the HMD, it is prone to small, high-frequent errors. To mitigate the translation of these errors to the end-effector's movements, a controller is employed, which defines the robot's dynamics when approaching the desired position.

Inverse Kinematics: Initially neglecting the translational movement of the robot part holding the needle, the robot's kinematics can be divided into horizontal and vertical components. In both dimensions, the robot features two joints The primary joint moves along a circular trajectory, the secondary joint rotates around the position of the robot on this trajectory (Fig. 7). This simplifies the inverse kinematics (IK) calculations to a circle-line intersection problem, as expressed by equations 1 and 2, referencing figure 7.

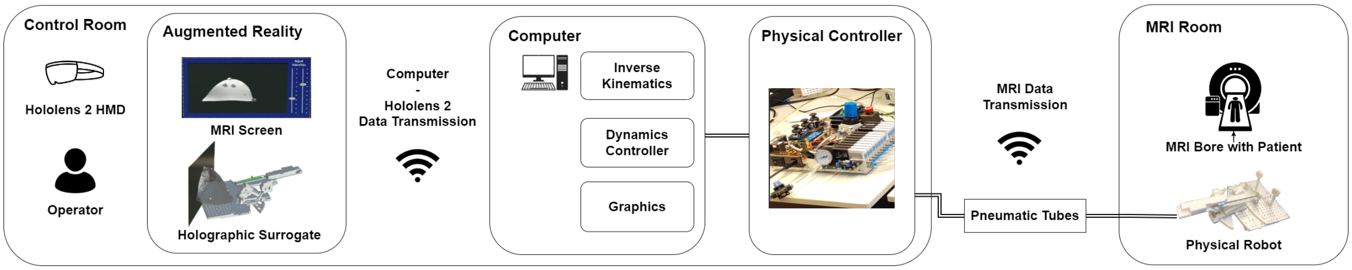


Fig. 5: The schematic overview of different components in the system. The operator is using the Hololens 2 HMD to visualize the scene inside the MRI bore. The IK, controller, and graphics calculations are done by the computer, which simultaneously communicates with the physical controller and the MRI scanner. The physical controller translates the stepper motor inputs to pneumatic inputs.

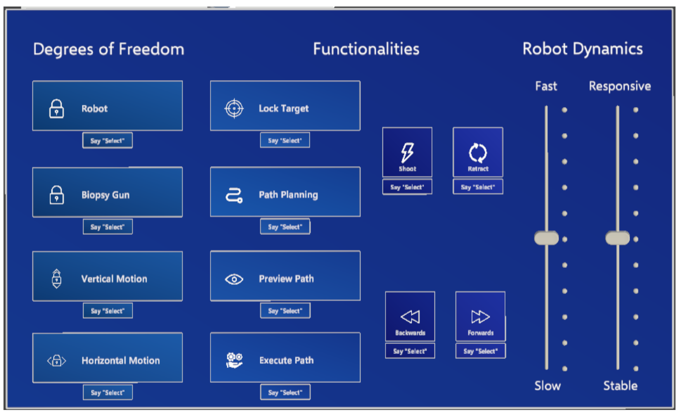


Fig. 6: The control panel, facilitating the operator to lock the DOFs of the robot, activate its functionalities, and adjust its dynamics.

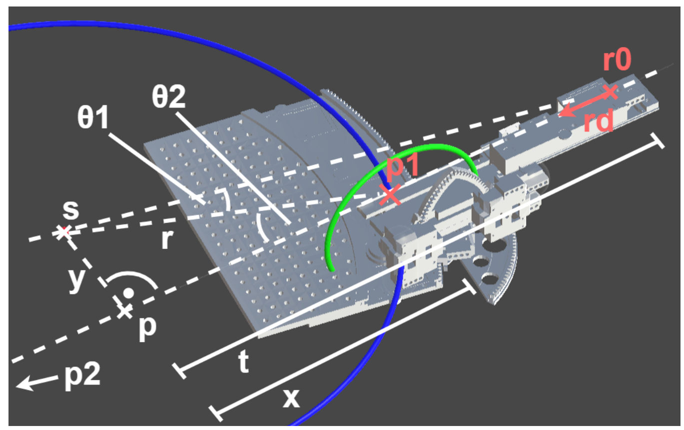


Fig. 7: The diagram visualizes the variables used in the IK calculations (Eq. 1 and 2). Here for the horizontal plane with the blue circle representing the trajectory of the robots joint. The green circle represents the same for the vertical plane.

$$y = s - (r_0 + \vec{r_d}((s - r_0) \cdot \vec{r_d}) \tag{1}$$

$$p_{1,\,2} = r_0 + \vec{r_d}(((s - r_0) \cdot \vec{r_d}) \pm \sqrt{r^2 - y^2}) \tag{2}$$

After calculating the intersection of the line representing the end-effector of the robot, with the circle representing the trajectory of the robot in that dimension, the target angle of the primary joint is determined by the angle that the intersection point forms with the baseline (Fig. 7). Subsequently, the angle of the secondary joint is defined by the angle between the trajectory of the end-effector and the projection of the intersection point onto the center of the circle.

A mapping of the joint angles to step positions of the robot is required (Eq. 3 and 4), where θ represents the joint angle and p the step position. d_{step} is a constant depending on the stepper motor parameters and the radius of the joint it moves along.

$$p_{\min} = \frac{\theta_{\min}}{d_{\text{step}}}, p_{\max} = \frac{\theta_{\max}}{d_{\text{step}}} \tag{3}$$

Mapping joint angle to step:

$$p_{\text{steps}} = p_{\min} + (\frac{\theta - \theta_{\min}}{\theta_{\max} - \theta_{\min}} * (p_{\max} - p_{\min})) \tag{4}$$

Control: To ensure precise control of the needle given the somewhat uncertain hand position estimation of the Head-Mounted Display (HMD), a straightforward Proportional-Derivative (PD) controller was implemented, rather than a direct one-to-one translation of movements (Eq. 5).

$$u(t) = K_p e(t) + K_d \frac{d}{dt} e(t) \tag{5}$$

Moreover, the motion of the hand was scaled down w.r.t. the desired change on the end effector's side, with a 2:1 ratio. For example, 20 cm of translation and 30 degrees of rotation of the hand would result in 10 cm of translation and 15 degrees of rotation of the desired position, respectively. Both the proportional gain K_p and the scaling ratio can be adjusted during runtime to tailor the dynamics to the operator's preference.

MRI Image to Robot Calibration

To register the relative position between the mesh created from the MRI images and the robot, a registration algorithm based on six MRI-detectable markers was developed. Through a connected component labeling algorithm, the positions of the markers from the 3D reconstruction of the

MRI data set are retrieved. The dataset is binarized using Otsu's method to determine a threshold between dark voxels representing empty space and light voxels representing the tissue and the markers [29]. Using a 3D kernel that scans the dataset, connections between neighboring voxels are established. The union-find algorithm is employed to find the root group for each connected component afterward [30]. The markers can be distinguished from other objects in the scene by their volume and shape. To transform the dataset into the correct position, orientation, and scale, the Procrustes method is applied [31]. This method assumes a correlation between the datasets, implying that a relationship between the detected marker array and their ground truth must be established. Distance metrics between the markers were used to identify the labels of the markers according to the ground truth.

Procrustes Analysis: Subsequently, the Procrustes analysis was performed to calculate the required scale, rotation, and translation to align the markers with the ground truth in the 3D space of AR [31]. For the integration into Unity, the Procrustes analysis was divided into three components and applied separately in the order of scale, rotation, and translation.

The scaling of the mesh to match the scale of the ground truth is determined by the ratio of the Frobenius norms between the two datasets centered around their centroids (Eq. 9).

$$||A||_F = \sqrt{\sum_{i=1}^{3}\sum_{j=1}^{n}|a_{ij}|^2} \text{ where } n = \#points \tag{6}$$

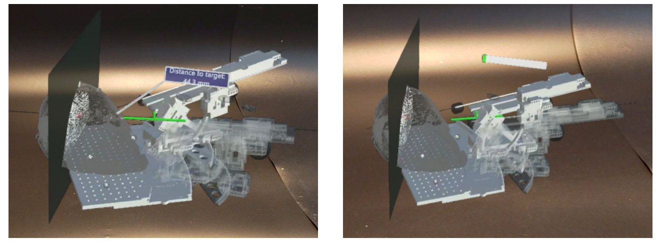
To determine the rotation between the datasets, the coordinates of each set are organized into $3 \times n$ matrices, where n is the number of markers. A singular value decomposition (SVD) is then performed on the matrix product of the datasets, from which the rotational matrix can be derived (Eq. 10). The resulting rotation matrix was transformed into a quaternion rotation for the application in Unity.

$$R = UV' \text{ where } SVD(Z) = U\Sigma V' \tag{7}$$

Finally, the translation is calculated as the vector between the centroids of the data sets.

Interactive Functionalities: The primary interactions with the holographic surrogate are facilitated through hand gestures. The operator forms a fist with their right hand to gain control of the desired position of the end-effector, which the surrogate, and subsequently the physical robot, follows with the defined dynamics. The same gesture with the left hand is employed to manipulate the position and orientation of the slicing plane in the mesh, displaying the corresponding MRI image on the screen in the Augmented Reality (AR) scene.

When the operator extends the index and middle finger of the right hand, they can designate a target on the MRI screen, which is subsequently transformed into the 3D space of the mesh. This target can be utilized by the auto-targeting function of the system. The auto-targeting mode removes the



(a) Planning the path with the phantom (transparent).

(b) The surrogate following the phantom's trajectory (opaque).

Fig. 8: The robot in the path planning mode. Defining a trajectory with the phantom (left - transparent), and following it with the surrogate (right - opaque). Here, the robot is also in the auto-targeting mode during the path planning (left), where the distance to the target is indicated by a blue label.

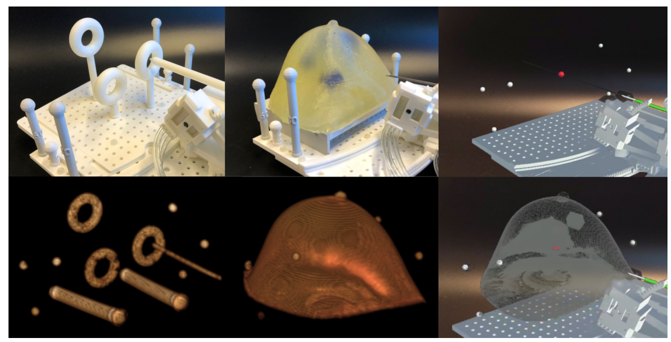


Fig. 9: (left) The targets and needle filled with petrolatum and their respective 3D reconstruction of the MRI images. (middle) The breast phantom and the respective 3D reconstruction. (right) The holographic surrogate of the robot with the marker ground truth, and after the import and automatic calibration of the MRI dataset of the breast phantom.

control of the robot's orientation from the operator's gesture control, allowing them to solely control the position of the robot. The orientation is calculated separately to maintain the alignment of the needle with the target. In the preview and execution mode during the path planning described later, the operator can either extend the index finger of their right hand or their left hand, to move the robot forward or backward along the trajectory, respectively.

A path planning function was implemented, creating a second "phantom" surrogate that operates independently of the physical robot (Fig. 8a). To distinguish the phantom from the opaque surrogate, it is depicted transparent. In this mode, the operator can control the phantom without the physical robot following its movements. After defining a path by moving the phantom to its desired position, the operator can switch to the preview mode. In this mode, the phantom can be moved back and forth along the defined path. Once satisfied with the path after the preview, the execution mode can be activated to enable the original surrogate, and consequently the physical robot, to follow the path until it reaches the phantom's position (Fig. 8b). Upon exiting the path planning mode, the operator regains manual control over

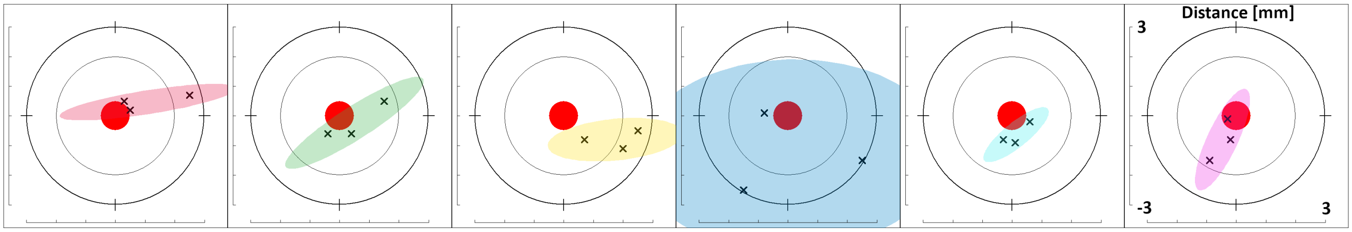


Fig. 10: The Euclidean distance to the three targets represented per trial.

the robot.

Experimental Evaluation: The calibration accuracy is assessed by calculating the target registration error (TRE) between the ground truth of the physical markers and their positions after the calibration.

To evaluate the performance of the AR-based control interface, two distinct experiments were conducted. In the first experiment, three torus-shaped targets were positioned at various heights and positions on the robot's base plate (Fig. 9).

The targets, as well as the mock needle used in this experiment, are filled with petrolatum, for visibility on the MRI images. To measure the accuracy of the targeting system, the intended targeting method of slicing through the imported MRI dataset and defining a target on the MRI screen is used. Subsequently, the auto-targeting function is utilized to bring the needle tip to the center of the torus-shaped target. Since earlier experiments showed warping in the 3D reconstruction of the MRI data perpendicular to the scan direction, which influenced the accuracy significantly (Sec. IV), two scans were used for targeting; a coronal scan for the horizontal plane, and a sagittal scan for the vertical plane. The distance from the needle tip to the center of the target is measured to quantify the accuracy of the targeting system. Since the biopsy needle extends for 17 mm when performing the biopsy, cutting away tissue along its path, an accurate positioning along its path is not paramount. For that reason, mainly the Euclidean distance of the needle trajectory was chosen for the evaluation of the accuracy of the system.

A second, qualitative experiment is conducted to simulate the intended application in breast biopsies. Two mock breast phantoms, with different stiffness, crafted out of softened PVC plastisol, with inlets of harder, colored material representing targets (e.g., lesions), are used to mimic the breast with lesions. The goal of this experiment is to verify the efficacy of the AR-based system in the assistance of the biopsy procedure, by successfully removing parts of the colored tissue in the breast phantom. The time taken for the different steps of the whole procedure is measured, as well as the success rate of extracting material from the target lesion.

III. RESULTS

The performance of the calibration algorithm was assessed through the TRE of twelve MRI scans. The mean TRE over all six markers in twelve scans was 0.86 $\pm$ 0.35 mm.

To identify potential biases in the targeting of the robot, such as the calibration system, both accuracy and precision were evaluated, the results are summarized in table I. The time taken to reach each target was also recorded. Using the previously described targeting method, while utilizing both coronal and sagittal scans combined, an accuracy of 1.36 $\pm$ 0.89 mm and a precision of 0.94 $\pm$ 0.60 mm could be achieved (Fig. I). It took a mean of 27.7 $\pm$ 2.9 seconds to reach the desired target. The distances of the needle trajectories to the targets per trial are visualized in figure 10.

TABLE I: The per trial accuracy and precision, as well as the average accuracy and precision, when targeting the torus-shaped targets.

Trial	**Accuracy** [mm]	**Precision** [mm]
1	1.24 $\pm$ 1.18	0.96 $\pm$ 0.40
2	1.01 $\pm$ 0.49	0.86 $\pm$ 0.43
3	1.97 $\pm$ 0.79	0.75 $\pm$ 0.32
4	2.21 $\pm$ 1.21	2.02 $\pm$ 0.39
5	0.80 $\pm$ 0.15	0.46 $\pm$ 0.18
6	0.96 $\pm$ 0.73	0.60 $\pm$ 0.30
Average	1.36 $\pm$ 0.89	0.94 $\pm$ 0.60

As initial experiments showed, that the 3D meshes reconstructed from the acquired MRI scans display significant deformations distal from its isocenter, the error w.r.t. distance of the targets to the isocenter was evaluated (Fig. 11). An ANOVA was performed to investigate the significance of the trend showing in the data, resulting in a p-value of 0.00107.

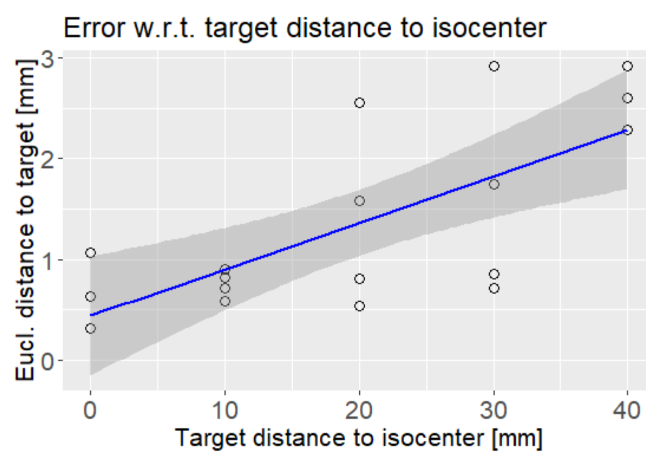


Fig. 11: The error w.r.t to the targets distance to the sagittal plane of the isocenter of the scan. The error significantly increases with increasing distance (p-value = 0.00107)

MRI-guided breast biopsies were simulated using the plastisol phantoms. Three biopsies were conducted inside the MRI bore on two different breast phantoms with different stiffness. The targets inside the phantom were cube-shaped and ranged in side length from 5 to 15 mm. Each of the in total six biopsies removed parts of the colored tissue, resulting in a success rate of 100% without any false negatives, albeit a small sample size (Fig. 12). The mean time spent on each biopsy, including acquisition scan, targeting, control scan and taking the biopsy was 11.83 ± 2.10 min.

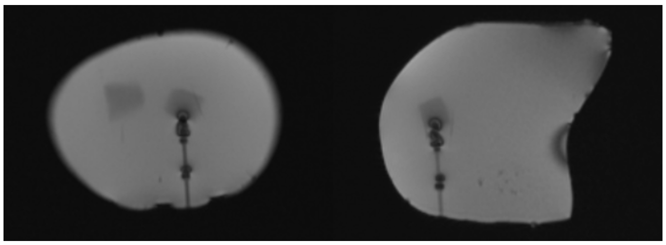


Fig. 12: One of the biopsy experiments, showing the coronal view (left) and sagittal view (right) of the breast phantom with the inserted needle at the target position. When taking the biopsy, the needle is extended by a further 17mm, cutting out tissue along this path.

IV. DISCUSSION

This study presents an MRI-integrated, accurate, and efficient AR-based method for conducting breast biopsies. The holographic surrogate enables the operator to intuitively control the biopsy robot through gestures, interfaces, and voice commands. Features such as visual cues, automatic targeting, and path planning assist the operator during the procedure, ensuring a fast and precise biopsy.

The results indicate that the system and the robot perform at a comparable level to current literature standards (Tab. II). Given that targeted tumors are rarely smaller than 5 mm in diameter, an accuracy of 1.36 ± 0.89 mm could be sufficient to conduct biopsies with a high success rate, especially when utilizing vacuum-assisted needles [8] and correcting for the identified causes of errors. This is also reflected in the results of the biopsy simulation, which achieved a 100% success rate, albeit a small sample size. The simulated biopsies took 11.83 ± 2.10 min. The current standard in clinical practise is around 35 to 39 minutes, excluding the preparation of the patient [8]. This shows, that the proposed system has the chance to significantly decrease the time the patient spends in the MRI bore, resulting in a less stressful experience. Moreover, a lower occupation time of the MRI room opens up the possibility for more control scans, covering a larger percentage of women with increased risk, which could increase the number of early detected lesions and in turn decrease the mortality rate.

TABLE II: Achieved accuracies in other biopsy robots.

	Proposed	**[11]**	**[12]**	**[13]**	**[14]**
Accuracy [mm]	1.36±0.89	1.04±0.15	0.70±0.04	0.34	7.4±4.6

The system demonstrating better precision than accuracy (Tab. III) suggests a bias in the system, potentially stemming from calibration errors or warping in the MRI images. Improving the calibration method, such as by minimizing potential rotations through different fiducial marker placements, could enhance accuracy and potentially achieve values closer to the precision.

During the evaluation of the performance of the system in an MRI environment, it became evident, that the largest cause for errors was introduced by a warping of the reconstructed 3D mesh from the MRI images (Fig. 14). The warping occurred perpendicular to the scan direction with which the data set was acquired. To circumvent inaccuracies introduced by this warping, the coronal and sagittal scans were both used for targeting, which increased the accuracy significantly (Tab. III). Figure 11 shows, that the accuracy is still significantly affected by the warping of the MRI images, as there is an increasing error with the distance of the target to the sagittal plane of the isocenter. Further investigation after the experiments made evident, that using the integrated 3D correction function of the MRI scanner would have mitigated this issue. This was evaluated by using a calibration cube with a grid of $11 \times 11 \times 11$ MRI detectable markers (Fig. 13). These results underline the previously mentioned effect of the distance to the isocenter on the accuracy (Fig. 11).

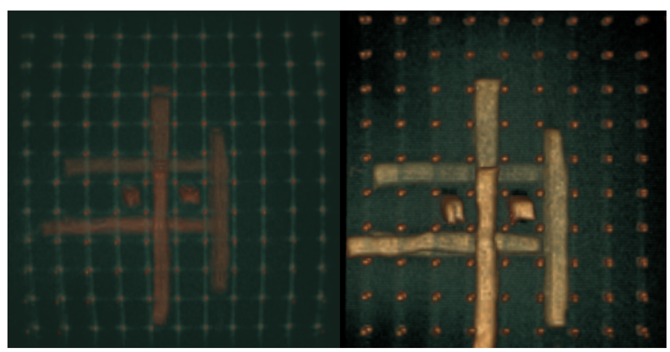
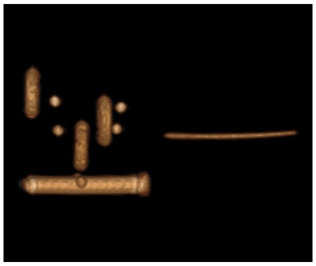
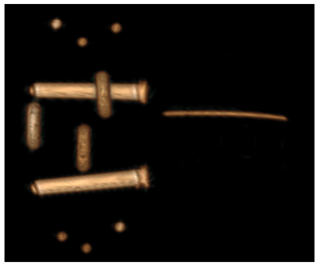


Fig. 13: The calibration cube used to determine the amount of warping after the experiments. Both images show the coronal view of the scan acquired in sagittal orientation. The result of using the MRI parameters used for the experiments are shown on the left side. The result of using the available 3D correction utility is depicted right.

The resulting accuracy can likely be expected to improve towards the values close to the isocenter, below 1 mm. Another solution would be to computationally correct the warping. Groenhuis et al. used a custom 3D calibration grid to construct a fifth-order polynomial correction function to counteract the warping [32].



(a) The 3D reconstruction of the coronal scan seen from the sagittal view.



(b) The 3D reconstruction of the sagittal scan seen from the coronal view.

Fig. 14: The 3D reconstruction from the MRI scan showing the view perpendicular to the scan direction. The line on the right side of each image is a petrolatum-filled mock needle, depicted as severely bent, although perfectly straight in reality.

TABLE III: The estimated marginal means (EMM) of the accuracy and precision when using either coronal or sagittal scan, and the accuracy and precision when using them combined.

	Accuracy [mm]	**Precision** [mm]
Coronal/Sagittal Scan	1.91 ± 1.00	0.68 ± 0.36
Combined Scans	1.36 ± 0.89	0.94 ± 0.60

Comparing the present findings with the previously published results (Sec. V) of the system's performance underscores the advantages of the implemented targeting system (Table IV). In contrast to targeting without the assistance of the targeting system, the accuracy witnessed a substantial improvement, increasing from 2.43 ± 1.32 mm to 1.36 ± 0.89 mm. The time required to reach each target was comparable to the other methods. It is crucial to note that in the prior results, the ground truth of the target positions was known, unlike the current method, where experimental outcomes relied on calibration. This underscores that despite the introduced calibration errors, the accuracy was significantly increased.

TABLE IV: The current results compared to those of the previous work [V], where the current targeting method was not yet developed.

Method	**Accuracy** [mm]	**Precision** [mm]	**Time** [s]
Current	1.36 ± 0.89	0.94 ± 0.60	27.7 ± 2.9
Manual	2.44 ± 1.32	1.64 ± 1.00	27.9 ± 6.4
3D-Target	2.72 ± 1.58	1.84 ± 1.39	37.7 ± 8.3
Joystick	3.71 ± 2.42	3.32 ± 2.30	29.0 ± 7.1

While the proposed system is currently operational, several limitations need to be addressed. The accuracy of the calibration method is constrained by the registration approach and the placement of markers. Exploring alternative marker placements and employing markers with enhanced detectability could improve calibration accuracy and, consequently, targeting accuracy. Various MRI-compatible calibration methods should be tested to optimize outcomes.

Presently, the system does not account for deformations of the breast during needle insertion. While a clamping system was used to mitigate this, deformations still occurred during the needle insertion. A model-based approach predicting target displacement during deformations should be considered. Real-time feedback is crucial to gather information and mitigate the impact of deformations.

Going forward, an integration with the MRI scanner to implement real-time feedback could be developed. APIs provided by MRI machine manufacturers enable the development of applications with direct access to the MRI scanner, facilitating the possibility of generating scanning planes along the needle trajectory. This could act as visual real-time feedback telling the operator the position of the needle relative to the target, as well as provide information about the deformation of the breast.

Given that the robot is driven by stepper motors, the accuracy is constrained by their step size. Optimizations, such as identifying joint configurations with minimal Euclidean distance from the needle trajectory to the target, could minimize errors introduced by stepper motors.

While the system is designed to be intuitive, new users require instruction on its functionalities and features, particularly when encountering AR for the first time. An integrated demonstration featuring animations and explanations could provide users with guidance without the need for additional personnel.

V. CONCLUSIONS

An AR-based system for control of an MR-safe robot to aid surgeons during MR-guided breast biopsies was developed to improve the targeting accuracy and decrease the occupation time of the MRI room. With an accuracy of 1.36 ± 0.89 mm, which was strongly affected by a correctable warping of the MRI images, it is comparable to the state of the art, while having the potential of reducing the MRI occupation time from 35-39 min to 11.83 ± 2.10 min. The system provides supporting functionalities such as auto-targeting, path planning, and automatic calibration, to enable a fast and accurate procedure. By facilitating the control of the robot through gestures, voice commands and interactive holograms, an intuitive human-robot interface was created. Going forward, the several identified error causes should be addressed to improve the performance of the system. Additionally, the implementation of real-time feedback through direct interfacing with the MRI machine could further improve the efficacy of the system.

References

[1] Sung H, Ferlay J, Siegel RL, Laversanne M, Soerjomataram I, Jemal A, Bray F. Global Cancer Statistics 2020: GLOBOCAN Estimates of Incidence and Mortality Worldwide for 36 Cancers in 185 Countries. CA Cancer J Clin. 2021 May;71(3):209-249. doi: 10.3322/caac.21660. Epub 2021 Feb 4. PMID: 33538338.

[2] Kashyap D, Pal D, Sharma R, Garg VK, Goel N, Koundal D, Zaguia A, Koundal S, Belay A. Global Increase in Breast Cancer Incidence: Risk Factors and Preventive Measures. Biomed Res Int. 2022 Apr 18;2022:9605439. doi: 10.1155/2022/9605439. PMID: 35480139; PMCID: PMC9038417.

[3] Siegel RL, Miller KD, Wagle NS, Jemal A. Cancer statistics, 2023. CA Cancer J Clin. 2023 Jan;73(1):17-48. doi: 10.3322/caac.21763. PMID: 36633525.

[4] Kuhl, C., Schrading, S., Leutner, C., Morakkabati-Spitz, N., Wardelmann, E., Fimmers, R., Kuhn, W., Schild, H. (2005). Mammography, Breast Ultrasound, and Magnetic Resonance Imaging for Surveillance of Women at High Familial Risk for Breast Cancer. Journal of clinical oncology : official journal of the American Society of Clinical Oncology. 23. 8469-76. 10.1200/JCO.2004.00.4960.

[5] Saslow D, Boetes C, Burke W, Harms S, Leach MO, Lehman CD, Morris E, Pisano E, Schnall M, Sener S, Smith RA, Warner E, Yaffe M, Andrews KS, Russell CA; American Cancer Society Breast Cancer Advisory Group. American Cancer Society guidelines for breast screening with MRI as an adjunct to mammography. CA Cancer J Clin. 2007 Mar-Apr;57(2):75-89. doi: 10.3322/canjclin.57.2.75. Erratum in: CA Cancer J Clin. 2007 May-Jun;57(3):185. PMID: 17392385.

[6] Rahman WT, Helvie MA. Breast cancer screening in average and high-risk women. Best Pract Res Clin Obstet Gynaecol. 2022 Sep;83:3-14. doi: 10.1016/j.bpobgyn.2021.11.007. Epub 2021 Nov 19. PMID: 34903436.

[7] Duffy SW, Vulkan D, Cuckle H, Parmar D, Sheikh S, Smith RA, Evans A, Blyuss O, Johns L, Ellis IO, Myles J, Sasieni PD, Moss SM. Effect of mammographic screening from age 40 years on breast cancer mortality (UK Age trial): final results of a randomised, controlled trial. Lancet Oncol. 2020 Sep;21(9):1165-1172. doi: 10.1016/S1470-2045(20)30398-3. Epub 2020 Aug 12. PMID: 32800099; PMCID: PMC7491203.

[8] Plantade, R., Thomassin-Naggara, I. (2014). MRI vacuum-assisted breast biopsies. In Diagnostic and Interventional Imaging (Vol. 95, Issue 9, pp. 779–801). Elsevier BV. https://doi.org/10.1016/j.diii.2013.12.023

[9] Feig S. Cost-effectiveness of mammography, MRI, and ultrasonography for breast cancer screening. Radiol Clin North Am. 2010 Sep;48(5):879-91. doi: 10.1016/j.rcl.2010.06.002. PMID: 20868891.

[10] Papalouka V, Kilburn-Toppin F, Gaskarth M, Gilbert F. MRI-guided breast biopsy: a review of technique, indications, and radiological-pathological correlations. Clin Radiol. 2018 Oct;73(10):908.e17-908.e25. doi: 10.1016/j.crad.2018.05.029. Epub 2018 Jul 2. PMID: 30041954.

[11] Song C, Yang Z, Jiang S, Zhou Z, Zhang D. An integrated navigation system based on a dedicated breast support device for MRI-guided breast biopsy. Int J Comput Assist Radiol Surg. 2022 Jun;17(6):993-1005. doi: 10.1007/s11548-022-02640-0. Epub 2022 Apr 30. PMID: 35489007.

[12] Liu W, Yang Z, Jiang S, Feng D, Zhang D. Design and implementation of a new cable-driven robot for MRI-guided breast biopsy. Int J Med Robot. 2020 Apr;16(2):e2063. doi: 10.1002/rcs.2063. Epub 2020 Jan 6. PMID: 31830358.

[13] Chan KG, Fielding T, Anvari M. An image-guided automated robot for MRI breast biopsy. Int J Med Robot. 2016 Sep;12(3):461-77. doi: 10.1002/rcs.1760. Epub 2016 Jul 11. PMID: 27402476.

[14] Vilanova JC, Pérez de Tudela A, Puig J, Hoogenboom M, Barceló J, Planas M, Sala S, Thió-Henestrosa S. Robotic-assisted transrectal MRI-guided biopsy. Technical feasibility and role in the current diagnosis of prostate cancer: an initial single-center experience. Abdom Radiol (NY). 2020 Dec;45(12):4150-4159. doi: 10.1007/s00261-020-02665-6. Epub 2020 Jul 23. PMID: 32705314.

[15] Moreira P, Patel N, Wartenberg M, Li G, Tuncali K, Heffter T, Burdette EC, Iordachita I, Fischer GS, Hata N, Tempany CM, Tokuda J. Evaluation of robot-assisted MRI-guided prostate biopsy: needle path analysis during clinical trials. Phys Med Biol. 2018 Oct 16;63(20):20NT02. doi: 10.1088/1361-6560/aae214. PMID: 30226214; PMCID: PMC6198326.

[16] Eslami S, Shang W, Li G, Patel N, Fischer GS, Tokuda J, Hata N, Tempany CM, Iordachita I. In-bore prostate transperineal interventions with an MRI-guided parallel manipulator: system development and preliminary evaluation. Int J Med Robot. 2016 Jun;12(2):199-213. doi: 10.1002/rcs.1671. Epub 2015 Jun 26. PMID: 26111458; PMCID: PMC4691445.

[17] Yang B, Roys S, Tan UX, Philip M, Richard H, Gullapalli R, Desai JP. Design, Development, and Evaluation of a Master-Slave Surgical System for Breast Biopsy under Continuous MRI. Int J Rob Res. 2014 Apr;33(4):616-630. doi: 10.1177/0278364913500365. PMID: 25313266; PMCID: PMC4193388.

[18] Desai JP, Gullapalli R. Image-guided robotic interventions for breast biopsy and ablative therapies: challenges and opportunities. Womens Health (Lond). 2009 May;5(3):229-33. doi: 10.2217/whe.09.1. PMID: 19392607.

[19] K. A. Stone, S. Dittrich, M. Luna, O. Heidari, A. Perez-Gracia and M. Schoen, "'1Augmented Reality Interface For Industrial Robot Controllers," 2020 Intermountain Engineering, Technology and Computing (IETC), Orem, UT, USA, 2020, pp. 1-5, doi: 10.1109/garIETC47856.2020.9249172.

[20] L. R. Islas, C. A. Paredes Orta and F. M. Chávez, "Forward and Inverse Kinematics Control of a Collaborative 6-DOF Robot for an Augmented Reality Educational System," 2021 International Conference on Electrical, Computer, Communications and Mechatronics Engineering (ICECCME), Mauritius, Mauritius, 2021, pp. 1-6, doi: 10.1109/ICECCME52200.2021.9590903.

[21] C. Xue Er Shamaine, Y. Qiao, J. Henry, K. McNevin and N. Murray, "RoSTAR: ROS-based Telerobotic Control via Augmented Reality," 2020 IEEE 22nd International Workshop on Multimedia Signal Processing (MMSP), Tampere, Finland, 2020, pp. 1-6, doi: 10.1109/MMSP48831.2020.9287100.

[22] M. E. Walker, H. Hedayati and D. Szafir, "Robot Teleoperation with Augmented Reality Virtual Surrogates," 2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI), Daegu, Korea (South), 2019, pp. 202-210, doi: 10.1109/HRI.2019.8673306.

[23] C. Thormann, M. E. Matour and A. Winkler, "Gesture Control of a Lightweight Industrial Robot Supported by Augmented Reality," 2023 27th International Conference on Methods and Models in Automation and Robotics (MMAR), Miedzyzdroje, Poland, 2023, pp. 406-410, doi: 10.1109/MMAR58394.2023.10242445.

[24] K. A. Szczurek, R. M. Prades, E. Matheson, J. Rodriguez-Nogueira and M. D. Castro, "Multimodal Multi-User Mixed Reality Human–Robot Interface for Remote Operations in Hazardous Environments," in IEEE Access, vol. 11, pp. 17305-17333, 2023, doi: 10.1109/ACCESS.2023.3245833.

[25] Z. Lin et al., "ARei: Augmented-Reality-Assisted Touchless Teleoperated Robot for Endoluminal Intervention," in IEEE/ASME Transactions on Mechatronics, vol. 27, no. 5, pp. 3144-3154, Oct. 2022, doi: 10.1109/TMECH.2021.3105536.

[26] C. M. Morales Mojica et al., "Interactive and Immersive Image-Guided Control of Interventional Manipulators with a Prototype Holographic Interface," 2019 IEEE 19th International Conference on Bioinformatics and Bioengineering (BIBE), Athens, Greece, 2019, pp. 1002-1005, doi: 10.1109/BIBE.2019.00186.

[27] J. D. Velazco-Garcia et al., "Evaluation of how users interface with holographic augmented reality surgical scenes: Interactive planning MR-Guided prostate biopsies," The International Journal of Medical Robotics and Computer Assisted Surgery, vol. 17, no. 5. Wiley, Jun. 08, 2021. doi: 10.1002/rcs.2290.

[28] H. Ranjan, M. van Hilten, V. Groenhuis, J. Verde, A. Garcia, S. Perretta, J. Veltman, F.J. Siepel, S. Stramigioli, "Sunram 7: An MR Safe Robotic System for Breast Biopsy". IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Oct 2023, pp. In press.

[29] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," in IEEE Transactions on Systems, Man, and Cybernetics, vol. 9, no. 1, pp. 62-66, Jan. 1979, doi: 10.1109/TSMC.1979.4310076.

[30] Bernard A. Galler and Michael J. Fisher. 1964. An improved equivalence algorithm. Commun. ACM 7, 5 (May 1964), 301–303. https://doi.org/10.1145/364099.364331

[31] Gower, J.C. (2010), Procrustes methods. WIREs Comp Stat, 2: 503-508. https://doi.org/10.1002/wics.107

[32] Groenhuis, V., Visentin, F., Siepel, F.J. et al. Analytical derivation of elasticity in breast phantoms for deformation tracking. Int J CARS 13, 1641–1650 (2018). https://doi.org/10.1007/s11548-018-1803-x

Appendix A
Calibration

For the operator to be able to accurately target a lesion with the system, the relative position of the breast to the robot has to be known. Several approaches can be taken to achieve such a calibration, with different levels of autonomy and different levels of freedom of the initial setup. A method with a low level of autonomy and a small degree of freedom would for example make assumptions about the initial position of the robot in the MRI bore and its joint positions. When these parameters are known, the position of the patient inside the MRI can be extracted from the MRI image meta-data and related to position of the robot. Such a method can work after an accurate initial calibration to find the position of the robot inside of the MRI machine. The lack of flexibility of this method likely outweighs the benefit of the trivial calibration procedure this creates. A change in the setup, e.g. where the robot is placed, renders this method useless until the initial calibration is renewed. A more generally robust and flexible method should be able to handle changes in setup and optimally also different joint positions of the robot, without the need for parameter changes in the algorithm. In this work, the focus was on solving the first issue of being able to reposition the robot relative to the target without restraining the functionality of the calibration method. Returning the robot's joints to their zero position is a trivial task, which is why the problem of having a calibration method considering different initial joint positions was postponed. For this reason, a more generalized approach was chosen in the form of the Procrustes method [31].

Mesh Creation

To perform the calibration, the DICOM data from the MRI scans has to be converted to 3D voxel data. The mesh creation of the 3D reconstruction of the MRI images is facilitated by the open-source volume rendering framework developed by Matias Lavik [7]. This framework is also employed to generate slicing planes, which are used to produce the corresponding MRI images Easy Volume Rendering.

Procrustes Method

In the Proctrustes method, two datasets, where each datapoint is related to a datapoint in the other dataset, are required. One dataset is then transformed to best fit the other. Example use cases of the method in the medical field are human pose estimation, gait evaluation, or coordination analysis [1], [2], [3]. In the case of this work, it is used to fit a set of markers to their known ground truth. Six MRI detectable markers are placed on the baseplate of the robot, in a way that is rotationally unambiguous (Fig. 15).

These show up in the MRI data as groups of bright pixels, or voxels in 3D, from which their relative position to each other can be derived. To get the positions of each group of voxels from the raw voxel data of the MRI images, a connected component labeling algorithm is applied.

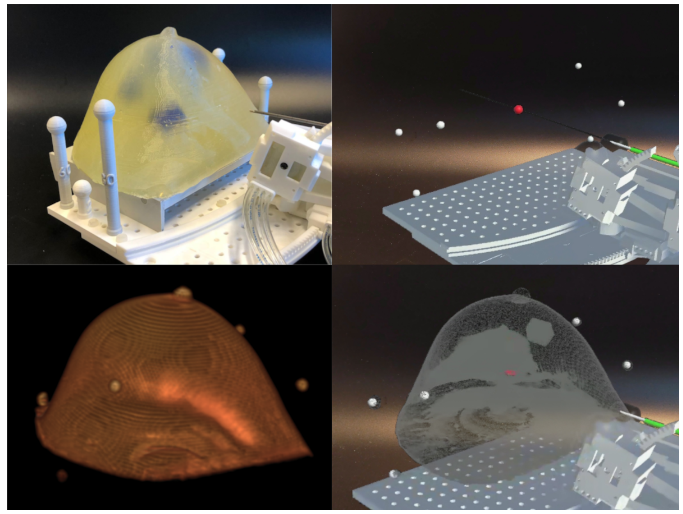


Fig. 15: (From left to right, top to bottom) The physical markers attached to the robot, their representation in the virtual world, how they show up on the MRI images and after the calibration.

Connected Components Labeling

Connected components labeling tries to solve the problem of finding groups in a dataset of adjacent components, for example, pixels in an image or voxels in a 3D dataset, without any information beforehand. It does so by scanning through the dataset identifying adjacent entities and grouping them with a label. Due to the consecutive order of scanning rows, and in 3D, layers, each group will consist of multiple subgroups that are interconnected (Fig. 16). A common root group will have to be found for those subgroups, in order to apply one label to the desired entity.

Thresholding: In order to be able to perform the connected component labeling, the dataset has to be binarized to distinguish voxels, that belong to groups from voxels that don't. A generalized solution to finding a binary threshold in a gradient dataset can be found in Otsu's Method [29]. If a histogram of the underlying grey-scale/intensity data exists, that has two clear peaks, a threshold in between the peaks can be found by this method. In the case of the MRI data, the two peaks represent the area with no tissue that shows up as voxels with low intensity, and the areas with tissue that show up as voxels with high intensity. The algorithm finds the threshold by minimizing the within-class variance, of the two classes, by calculating the weighted sum of variances (Eq. 8).

$$\sigma_w^2(t) = \omega_0(t)\sigma_0^2(t) + \omega_1(t)\sigma_1^2(t) \quad (8)$$

After the binarization of the dataset, the kernel which is used to scan through the dataset and connect the voxels has to be defined. A three-dimensional dataset requires a three-dimensional kernel, where multiple types of kernels with varying complexity, and in turn accuracy, can be used. In this application, the simplest kernel was sufficient, consisting of

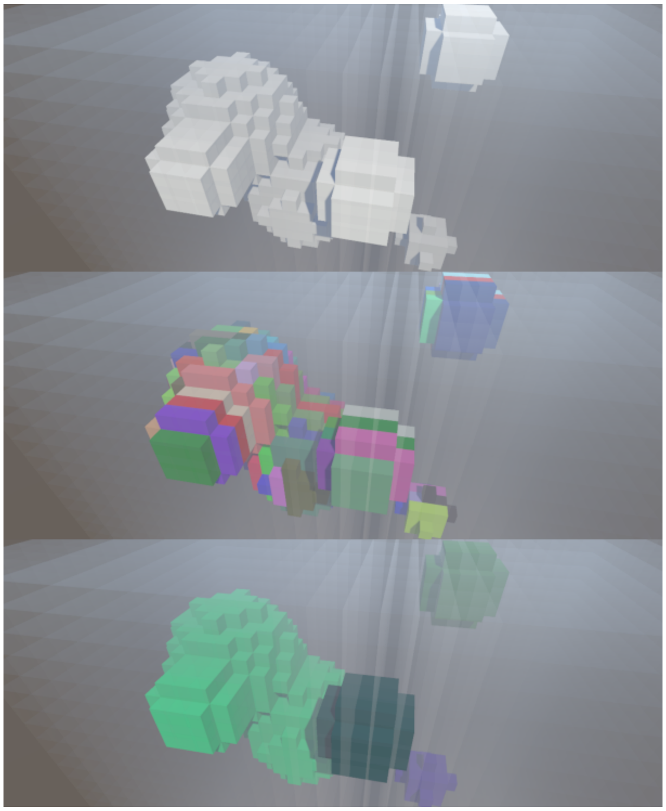


Fig. 16: The connected components before labeling, labeled with subgroups, and finally labeled into the target groups after the union find algorithm was applied. An abstract example on the use of the connected component analysis. Here four different groups have to be identified.

three voxels around its center, reaching in each dimension (Fig. 17).

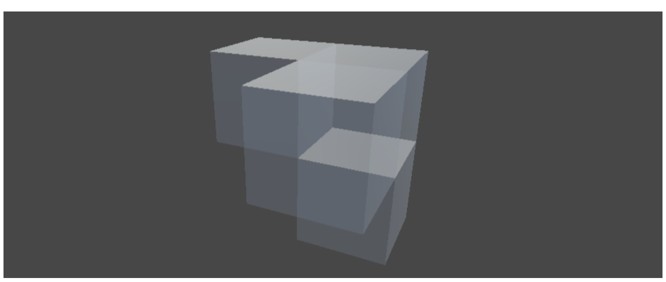


Fig. 17: The kernel used in the connected components labeling algorithm, scanning from left to right, bottom to top, front to back.

Scanning through the dataset row by row, the kernel checks if the voxel below it has a label, if it does, it takes on this label, if it doesn't it checks the same for the voxel to its left and takes on that label. If both cases are not true, a new label is created. To make connection in the third dimension, the kernel also checks, if there is a voxel in front of it in the previous layer, if there is, it makes its label the parent label of the current label. This results in a lot of subgroups inside the components, which are interconnected through a parent-child relationship (Fig. 16). Finding a common root group can then be done via the union-find algorithm.

Union Find Algorithm: The Union Find algorithm finds a common root for disjoint sets [30]. During the labeling procedure, the subgroups are inter-connected, but initially, a common root is not obvious, as no clear hierarchy in the parent-child relationships is evident, and recursive relationship can exist (Fig. 18).

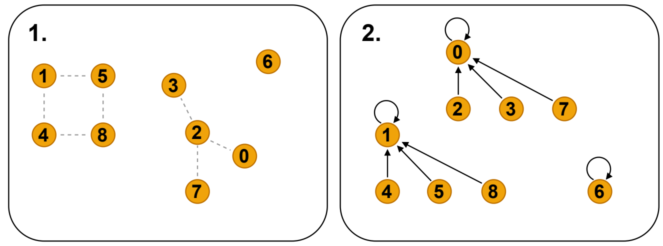


Fig. 18: An illustration of the union find algorithm. A single root is found for the connected labels, even if a recursive relationship is present.

The union find algorithm merges these overlapping, smaller sets and creates a tree of connections. Then it finds a common root, even when the parent-child relationships are recursive. In code, the algorithm can be implemented via a class with according members. Listing 1 shows an example implementation of such a label class [4].

Listing 1: An example implemetation of the Union Find Algorithm [4].

```
internal class Label
{
  public int Name { get; set; }
  public Label Root { get; set; }
  public int Rank { get; set; }

  public Label(int Name)
  {
    this.Name = Name;
    this.Root = this;
    this.Rank = 0;
  }

  internal Label GetRoot()
  {
    if (this.Root != this)
    {
      this.Root = this.Root.GetRoot();//Compact tree
    }
    return this.Root;
  }

  // The Union Find Algorithm.
  internal void Join(Label root2)
  {
    if (root2.Rank < this.Rank)//is the rank of Root2 less than that of Root1 ?
    {
      root2.Root = this;//yes! then Root1 is the parent of Root2 (since it has the higher rank)
    }
    else //rank of Root2 is greater than or equal to that of Root1
    {
```

```
      this.Root = root2;//make Root2 the parent

      if (this.Rank == root2.Rank)//both ranks are equal ?
      {
        root2.Rank++;//increment Root2, we need to reach a single root for the whole tree
      }
    }
  }
}
```

After all connected components are labeled with a single label, the amount of voxels contained in each component can be used to determine their volume. The volume per voxel can be derived from the meta-data of the MRI image, containing resolution and slice thickness. This volume can then be multiplied by the amount of voxels in the component. Similarly, the coordinates of the voxels contained in each component can be averaged to determine its position. Using this, the six markers and their position in voxel space, meaning their relative placement to the number of voxels in the 3D cube, can be extracted. To be able to use this information, the positions have to be converted into a common space with the ground truth. The world space in the Unity game engine is used for that purpose, the ground truth is converted from the space of the base plate of the robot to world space, the connected components from voxel space to world space.

With both datasets in the same space, each marker from the connected components set has to be linked to the corresponding marker of the ground truth, in order to proceed with the calibration. An algorithm was developed that automatically establishes this connection, given some restrictions concerning the positioning of the markers relative to each other.

Closest Pair Correspondance: The algorithm operates under the assumption that the distances between each data point and its two closest other data points are unique and that the transformation between the datasets involves only translation, rotation, and scaling. For each point in both datasets - the ground truth and the detected components - the distances to every other point are calculated and sorted. The algorithm identifies the two data points in each dataset with the smallest distance to the next data point (Fig. 19). To determine which of the two marker points corresponds to which of the two ground truths, the second smallest distance to the next marker is considered. The point with the smallest distance to the second-closest marker corresponds to the ground truth with the second-smallest distance to the second-closest ground truth. By removing these two correlated points from the dataset and repeating this process until none are left, a relationship between each marker and the ground truth can be established.

Once a connection of all the detected markers from the MRI images to the ground truth is established, the Proctrustes method can be applied to transform the dataset to match the position, orientation, and scale of the ground truth, effectively resolving the calibration (Alg. 1).

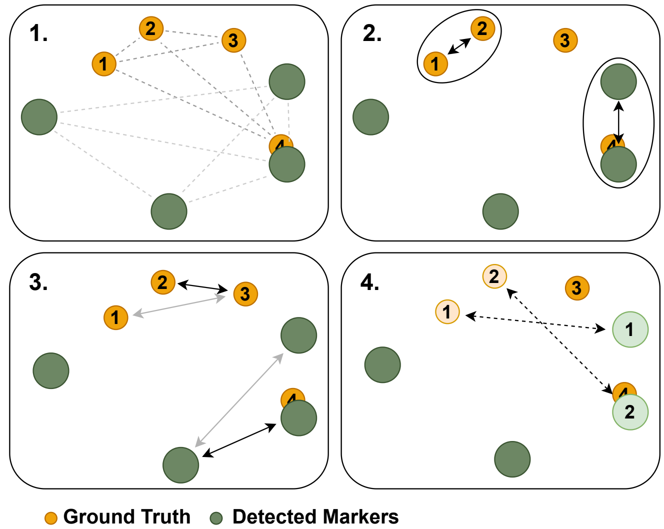


Fig. 19: An illustration of the closest pair correspondence algorithm. (1) For both the known and unknown datasets, the distances between the data points are calculated. (2) The data pair with the closest distance is identified. (3) The distance to the second closest data point is evaluated to distinguish the labels of the pair. (4) The points of the unknown datasets are labeled, then the closest pairs of both sets are removed from the algorithm.

Algorithm 1 Closest Pair Correspondance

Dictionary correspondanceMap;
List[List] distances = GetDistancesToOtherMarkers(data, groundTruth);
while # Data Points $>$ 1 **do**
  closestPairs = GetClosestPairs(distances)
  secondClosest = SecondClosestDistance(closestPairs)
  correspondanceMap[groundTruth] = data;
  RemoveMappedDataPoints();
end while

Procrustes Analysis: The Procrustes analysis uses the singular value decomposition (SVD) to apply a transformation of one dataset, that has a connection to another, to find the optimal orthogonal linear transformation to match this dataset. For easier application to the Unity game engine, the components of the Proctrustes analysis were split and ordered into scaling first, rotation second and translation last (Fig. 20). Scaling was achieved by multiplying the dataset to be scaled by the ratio of the Frobenius norms of both the dataset and the ground truth, centralized around their centroid (Eq. 9).

$$||A||_F = \sqrt{\sum_{i=1}^{3}\sum_{j=1}^{n}|a_{ij}|^2} \text{ where } n = \#points \tag{9}$$

The needed rotation was calculated by performing an SVD on the matrix product of the marker coordinates of the two

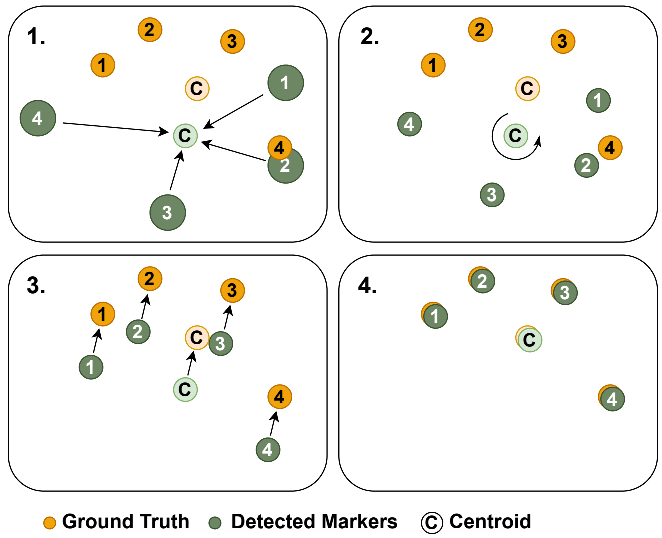


Fig. 20: An illustration of the transformation done via the Procrustes analysis. (1) The Frobenius norm centered around the centroid of each data set is calculated. The ratio of the Frobenius norm is used to scale the detected dataset to the size of the ground truth. (2) The SVD of the matrix product of the two datasets is used to calculate the rotation between them, here the corresponding labels are crucial. The resulting rotation matrix in converted to a quaternion rotation and applied. (3) The detected dataset is transposed along the vector between its centroid and the centroid of the ground thruth. (4) The detected dataset is successfully aligned with the ground truth, completing the calibration.

datasets. The necessary rotation matrix is then derived by multiplying the left singular vectors with the right singular vectors (Eq. 10). For use in the Unity game engine, the rotation matrix has to be converted to a quaternion rotation (Eq. 11).

$$R = UV' \text{ where } SVD(Z) = U\Sigma V' \tag{10}$$

$$\begin{aligned} q_w &= \frac{1}{2}\sqrt{\max(0, 1 + m_{00} + m_{11} + m_{22})} \\ q_x &= \frac{1}{2}\sqrt{\max(0, 1 + m_{00} - m_{11} - m_{22})} \\ q_y &= \frac{1}{2}\sqrt{\max(0, 1 - m_{00} + m_{11} - m_{22})} \\ q_z &= \frac{1}{2}\sqrt{\max(0, 1 - m_{00} - m_{11} + m_{22})} \end{aligned} \tag{11}$$

An additional check for the correct sign of the quaternion has to be made in order to have a consistent representation (Eq. 12).

$$\begin{aligned} q_x &= q_x \cdot \text{sign}(q_x \cdot (m_{21} - m_{12})) \\ q_y &= q_y \cdot \text{sign}(q_y \cdot (m_{02} - m_{20})) \\ q_z &= q_z \cdot \text{sign}(q_z \cdot (m_{10} - m_{01})) \end{aligned} \tag{12}$$

Finally, the translation was calculated by determining the centroids of the ground truth, and the scaled and rotated dataset, then moving the dataset along the vector between the centroids.

APPENDIX B
SOFTWARE ARCHITECTURE

As software grows in complexity over the time of development with added features and control modes, the complexity of the code grows exponentially. Parts of code start to depend on others which themselves depend on other parts of code, forming a complex dependency structure, which slows down development. Adding new features becomes harder, as weaving them into the net of dependencies without breaking the whole fabric of the rest of the code becomes more and more difficult.

Similarly, a greater complexity hinders the performance of the program, necessitating optimizations to ensure a high enough frame-rate to be comfortably operatable. Since most of these things are not evident or predictable from the start of the development, it is almost guaranteed, that the initial state of the program will be sub-optimal in a lot of places. In software development, these shortcomings are called ”technical debt”. The term implies, that these problems will have to eventually be tackled in order to not cause bigger issues in the long run. The process of restructuring and optimizing the program is called ”refactoring”. Over the development of the software described in the paper, the code was refactored multiple times to address the technical dept introduced by the quick development in earlier stages of development.

In object-oriented programming (OOP), which the language C# as used for Unity development counts to, software design patterns resolving common issues causing technical dept were developed for over 30 years [5]. These structures and ideas, while undergoing improvements and reformations, remain highly relevant and apply to almost all software utilizing OOP. Several of these ideas and abstractions were applied in this work to ensure the performance of the program, as well as to ensure that further development is possible without hindrance.

State Machine

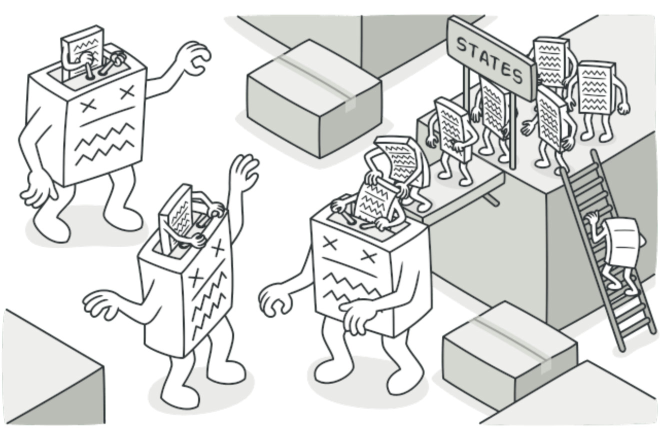


Fig. 21: An illustration of the state machine - the interchangeable states dictate the behaviour of the program [6].

One way of optimizing the performance of the program is to ensure, that only the code necessary at any given moment, is running. A state machine splits the logic into parts, or states, that are only being executed when the according state is active (Fig. 21). Through certain conditions or inputs, the state machine can switch states to change to the expected behavior. This is achieved by implementing an abstract ”base-state”, that the other states are inheriting from, or a ”base-state” interface, that the other states implement (Fig. 22). This ensures, that all states include the same methods, such as ”execute” or ”check switch state condition”. The term ”method” refers here to a function inside of a class. The methods themselves are implemented differently for every state according to its desired behavior. This creates an abstraction layer, where each state looks the same from the ”outside”, but differently from the ”inside”. The state machine itself then only holds the context - or variables needed in the states - and a reference to the currently running state. Conversely, each state holds a reference to the state machine, to be able to access the context and variables. The state machine only executes the abstracted methods which every state has implemented making it autonomously run by itself given the correct conditions for switching between the states.

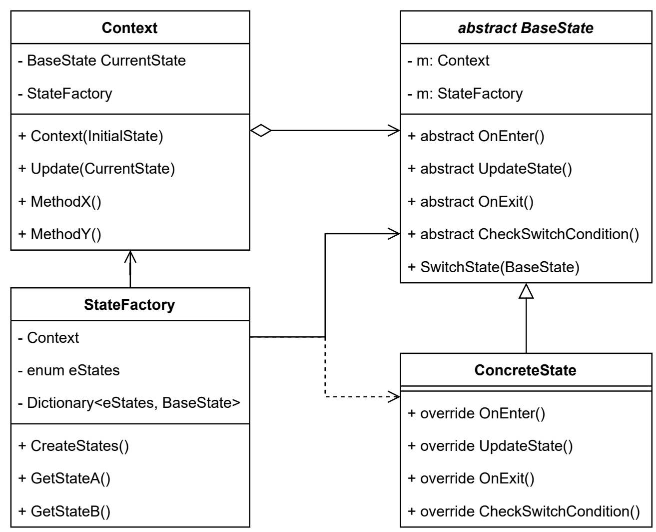


Fig. 22: The class diagram of the implemented state machine and it’s factory.

This design was applied to most parts of the program requiring some kind of control - the surrogate and its phantom, the slice generating the MRI images, and the ray indicating the desired trajectory of the end-effector of the robot. The slice required the simplest implementation of the state machine, an idle state, and a controlled state. In the idle state simply the switch condition is checked, other than that, no code is executed. Once the operator forms a fist with their left hand, the ”controlled state” is entered, facilitating control over the slice by moving and rotating the fist. A controller was implemented to define the behavior between gesture and slice. Similarly, the state machine for the ray was implemented, with the addition of two targeting states, one idle and one controlled. The targeting state ensures, that the ray is always pointing towards the target in 3D space which was defined by the operator. In that state, the operator is able to control the position of the robot, whereas the rotation is automatically decided by the targeting state. This creates the necessity for an extra idle targeting state, as the code for the position control does not need to be executed when the operator releases the control gesture, but the rotation should still be automatically adjusted, if the target changes position. As a result, the target can be moved around in space, while the robot's end-effector keeps alignment with it.

Hierarchical State Machine

A more sophisticated state machine had to be implemented for the virtual surrogate of the robot and its phantom (Fig. 23. This state machine encapsulates two different entities in one state machine, which increases its complexity. This added complexity is tackled with a hierarchical state machine. Instead of adding dozens of situational states, a hierarchy of a few main- and sub-states is introduced. The parent states function similarly to the states described earlier, with the difference of them delegating functionalities to their sub-states. The sub-states have in turn the authority to conditionally change states between each other, the parent-state is detached from this logic. Additionally, the amount of conditional state-change checks each state has to make each frame reduces significantly. This simultaneously makes the program more efficient, as well as more comprehensible for future developers.

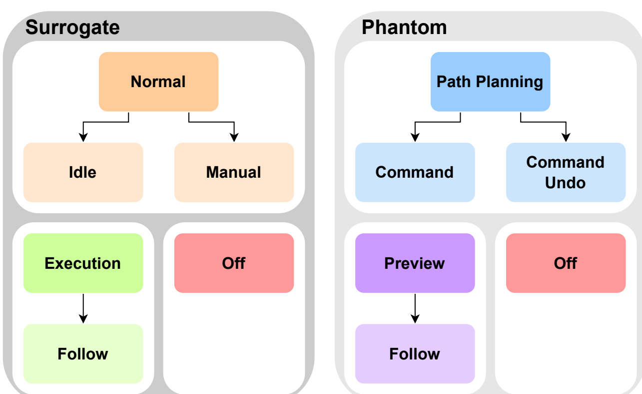


Fig. 23: The structure of the hierarchical state machine implemented in this work.

In this application, the main states are responsible for the general mode the robot is in. This is either the manual controlled mode, the path planning mode, the preview mode, or the execution mode. Additionally, the virtual robots can be completely turned off. The different modes also dictate, which version of the virtual robot - surrogate and phantom - is active, or even visible. The Surrogate can be in its normal, manually controlled mode, where the phantom is not visible, and in its execution mode, in which the phantom was previously used to define a path that is now followed. The phantom is visible in this mode, but inactive. The phantom in turn is active in the path planning mode, used to define the path for the surrogate to execute, while the surrogate itself is inactive. Similarly, the surrogate is inactive during the preview mode, in which the phantom moves along its own defined path. The basic functionalities are delegated to the sub-states, which mostly are responsible for the motion of the robot. A special case exists in the path planning state of the phantom, where a command pattern was implemented to save its trajectory to be later executed by the surrogate and in turn the physical robot.

Command Pattern

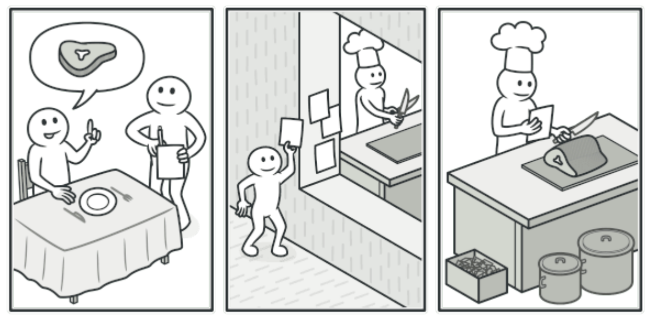


Fig. 24: A real-world analogy of the commend pattern - the chef receives orders and processes them in succession [6].

The problem of path planning was solved by the creation of a phantom of the virtual surrogate, which can be controlled like the surrogate itself, while not causing the physical robot to follow its motions. Its path taken during these movements is saved in a reference list, which can be reused to make the phantom, and later the surrogate move along it, both forward and backward, while the movement can be initiated from any position in the list. This is done with the command design pattern (Fig. 24, which consists of a command interface and a command invoker class (Fig. 25). The command interface contains an "Execute()" and an "Undo()" method, whose implementation is defined in concrete command classes. In the context of the path planning mode, the concrete command class "MovementCommand" executes the movements of the robot through the controller aligning the end-effector with the ray with the defined dynamics. For each increment in the movement of the phantom, a new object of the MovementCommand class is created and stored in a list. Each MovementCommand contains the Execute() and Undo() methods as well as the position before and after the incremental movement. This way, the Execute() method can move the position of the robot to the next in the list, while the Undo() method can do the inverse movement to move backward along the path.

The command invoker is responsible for storing the list containing the MovementCommand objects and adding and removing MovementCommands from the list. In the specific application, it is also responsible for checking if a movement even occurred. This is useful, as the operator can decide to pause during the path planning, which would add periods of non-motion to the path. These periods of pauses are not

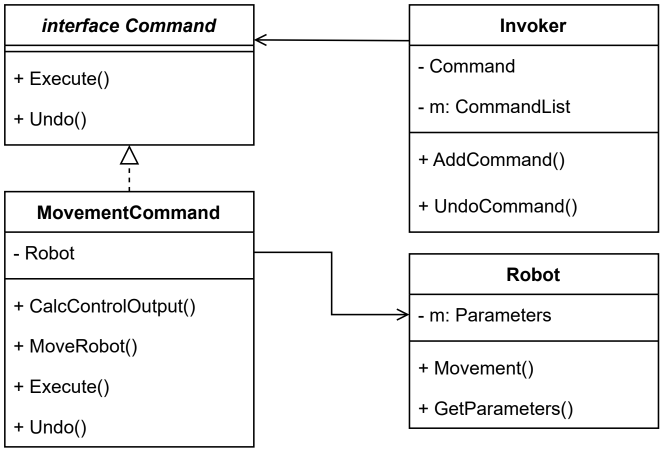


Fig. 25: The class diagram of the command pattern, with the MovementCommand as an example.

desirable in the later execution of the path and, thus are not added to the list, essentially creating one smooth path, even if pauses are taken.

In the preview and execution mode, the phantom, and surrogate respectively are moving along this path by calling the Execute() and Undo() methods of the MovementCommand objects in the list and moving to the next or previous object in the list.

Factory

In the context of the state machines, the abstraction of the implemented base-state interface demands an abstraction of the reference to the state it switches into. The objects using the base-state interface only know that they will have to receive some form of state to turn into, but not what specific state. For this problem a factory design pattern is useful. It counts to the creational patterns, as the name implies, it will create the states used in the state machines. The typical factory class creates and returns a specific demanded state of the type "base-state", which means only a reference to the factory is needed to have access to all states. The slightly, in terms of performance, improved implementation of the factory used in this work takes a different approach. Upon creation of the factory, each available state is created and its reference is stored in a dictionary, that can be accessed through a enum stating the type of desired state. The advantage is, that each time a state is demanded of the factory, it does not have to create a new one, but merely passes the reference to the initially created state. This one-time creation of each state should increase the performance of the state machine.

Mediator

The growing complexity of the software introduces cross-dependencies between the components of the program. In the presented application for example, the robot depends on the ray, which depends on the voice commands, which depend on the state of the UI menu, which depends on the user input handler, the UI menu and user input handler

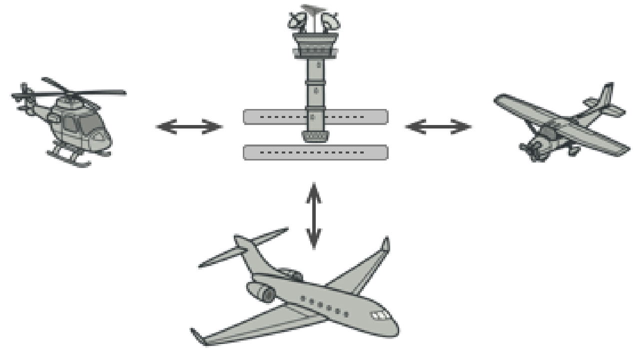


Fig. 26: A real-world analogy of the mediator pattern - aircraft pilots communicating through the control tower [6].

also depend on the voice commands, and so on. This net of dependencies grows and gets tighter each time a new component is introduced into the system. Resolving these dependencies is key to ensure that further development can be made without a lot of resistance. Similarly, changes in one part of the program would affect all other parts depending on it.

The mediator design pattern was used to tackle this issue (Fig. 26). Instead of the complex net of dependencies between the components, a separate class acts as a mediator, hence its name. As a result, each component only depends on the mediator instead of all the other components.

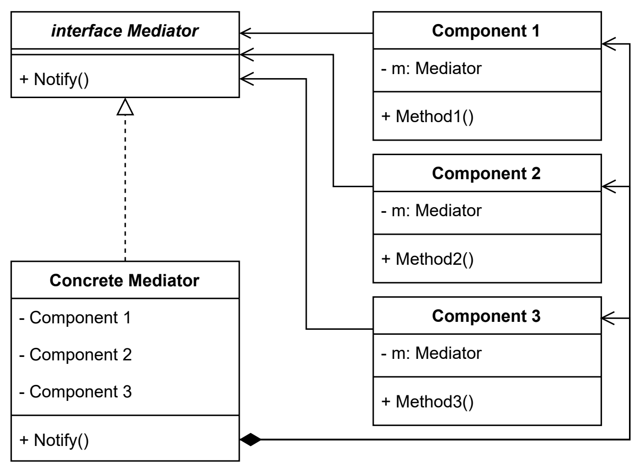


Fig. 27: The class diagram of the mediator pattern.

As for the implementation, a mediator interface containing a "Notify()" method is made, which handles all the communication traffic (Fig. 27). The concrete mediator implements this Notify() method to define the behavior of how each notification is handled. A class holding a reference to that interface called "MediatorBase" is created, which is inherited from by all the components communicating through the mediator. To initially set the reference to the concrete mediator in all the components, an evoker is used. It holds references to all components and creates the concrete mediator object, making all components member variables of the concrete mediator. Upon construction, the concrete mediator then passes itself as a reference to all the components. To communicate, the components can now call the Notify() method of the concrete mediator, in which they simply pass a string describing its intention and a reference to themselves, for the case in which it is relevant which component sent the notification. In the Notify() method, the concrete mediator then defines which component has what reaction to the notification that was sent. An added benefit to this setup is, that the same functionality of a component can be triggered from many different angles by simply passing the according notification. For example, if the operator wants to lock the degrees of freedom of the robot, they can either click a button in the UI, which sends the appropriate notification, or trigger a voice command, which sends the same notification, the reaction will be the same. Adding another angle to lock the robot, for example through a gesture, now becomes trivial.

Appendix C
Control Paradigm and Functionalities

Control Paradigms of the Software

Gesture Recognition: Through the outside cameras of the HMD, the hand of the operator is parameterized into 25 components representing the links of the fingers and dividing the palm into its main components (Fig. 28). The relative position of the finger links to each other and the palm can be used to calculate the curl of each finger, ranging from 0 to 1 representing fully extended fingers and fully curled fingers respectively. A finger is classified as curled if it satisfies a curl threshold of ≥ 0.5, and as extended, if it satisfies a curl threshold of ≤ 0.1. This metric is used to define the gestures used in the system by applying these thresholds on each finger separately. A fist gesture is recognized, when all fingers are classified as curled, a two-finger gesture is recognized, when index and middle fingers are classified as extended and the rest as curled, a one-finger gesture is recognized if only the index finger is extended. Another metric utilized from the parameterization of the hand is the distance of the links to each other. The distance can be used to determine if fingertips touch each other. In the case of this application, a double tap of the index finger and thumb is used to activate certain functionalities. This gesture is defined by the distance of the fingertips crossing a certain distance threshold three times in a period of 0.5 s.

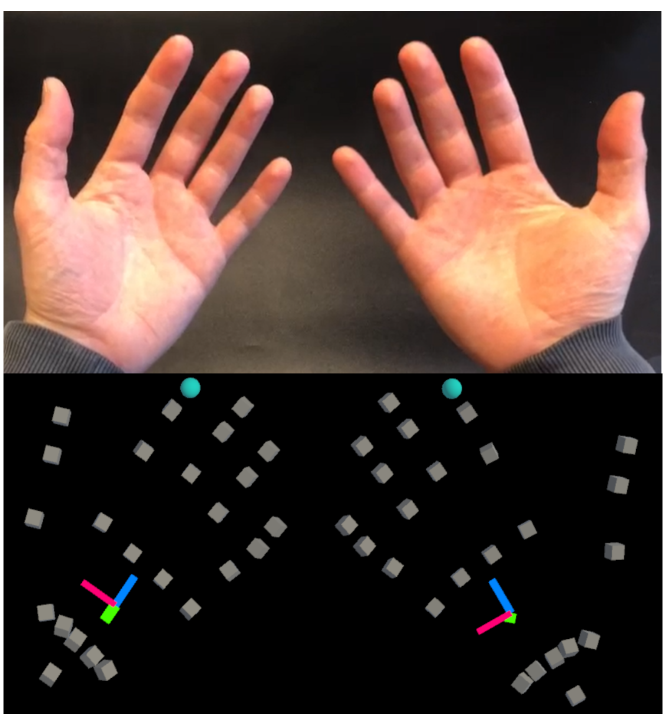


Fig. 28: The cuboids representing the parameters of the fingers as seen from the users perspective, here from a palmar view.

Moving the robot: The operator gains control over the robot by forming a fist gesture. The ray then follows the position and rotation of the fist, taking into account its relative position in 3D space compared to where the fist was initially formed (Fig. 29). This design is crucial to ensure that ray control remains independent of the operator's position and hand orientation relative to the robot. In practical terms, this means that the holographic robot mimics movements from the operator's perspective, rather than w.r.t. its own frame. This approach significantly enhances control intuitiveness, eliminating the need for the operator to consider the robot's frame relative to the world frame during operation.

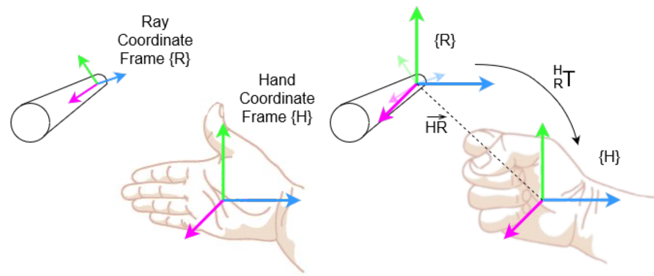
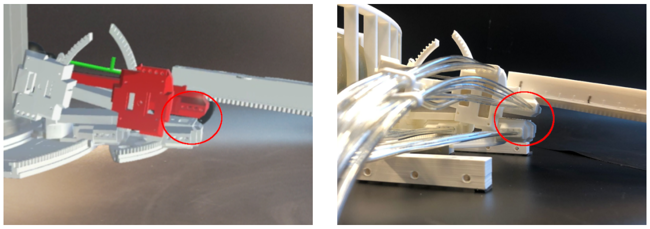


Fig. 29: A schematic of how the gesture control of the ray is facilitated, which defines the desired end-effector position.

Locking Degrees of Freedom: Through either voice commands or the control panel, the operator has the capability to both lock and unlock specific degrees of freedom (DoF) of the robot. Locking can be applied to horizontal or vertical movement independently, simplifying adjustments along a single dimension. Additionally, the operator can opt to lock the robot's movement in all dimensions, this feature is mainly used when the desired position and orientation for needle insertion have been achieved. By default, the robot's end-effector is locked along its sliding joint, but the operator can unlock it at any point, primarily for performing the needle insertion during the procedure.

Collisions: The robot has the inherent risk of self-collision, which could result in the stepper motors skipping steps and consequently disrupt the program's ability to track the robot's position. Given the MRI environment, encoders can't be utilized. To avert collisions within the robot, such as between the needle and the baseplate, a collision prevention script was developed. This script detects when the robot is approaching a potential collision and locks the robot's movements that could lead to such collisions. This collision prevention mechanism is versatile and can be extended to safeguard against collisions with any object present in the virtual space, including protecting against collisions with the patient or the MRI machine.

Auto-targeting: The auto-targeting mode constitutes the most important and most used feature of the system. When activating this mode, the control over the orientation of the ray defining the desired end-effector position and orientation is released from the operator, which only retains control over the end-effector position. The ray is forced to keep its orientation towards a target in the 3D mesh, whose position is defined by the operator. The operator does so by using a two-finger gesture with the index and middle finger, with

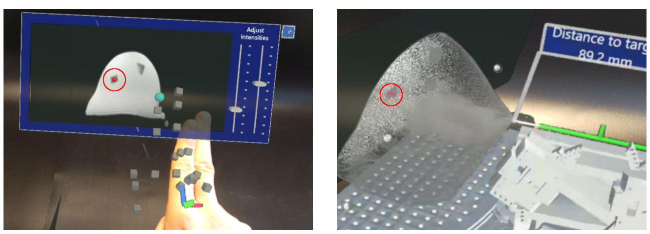


(a) The collision being detected and prevented in the virtual space.

(b) The resulting position of the real robot shortly before the collision.

Fig. 30: An example of a prevented collision. The part subject to a potential collision lights up red (a) and locks its motion in the direction of the collision. The collision prevention is translated to the real robot (b).

which they gain control over a target on the MRI screen (Fig. 31a). This target is then converted from the 3D MRI image into a 3D position in the mesh through the slice position inside of the mesh (Fig. 31b). Since the orientation of the ray is constantly updated in this mode, the robot will follow the target in real-time. This could be exploited to keep orientation to a moving target, for example, due to respiration.



(a) The operator defining the target on the MRI screen.

(b) The target translated into it's position in the 3D mesh.

Fig. 31: The interaction with the system in the auto-targeting mode. The operator defines a target on the MRI screen by using the two-finger gesture (a). This target is translated into the 3D space of the mesh for the robot to target (b).

Moving and resizing the hologram: To relocate the hologram to a specific position, users can activate the moving and resizing mode, either as a voice command or via the main menu. This mode introduces a bounding box surrounding the hologram, which can be manipulated by pinching gestures (Fig. 32). This enables users to effortlessly move the hologram to the desired position, as well as rotate and resize it to their preferred dimensions. It's important to note that these adjustments do not impact the robot's operation, as all functionalities account for the changes in scale introduced during this process.

User Interface

Addition to Inverse Kinematics from the paper

The resulting joint angles from the IK have to be converted to step positions of the stepper motors. The conversion depends on the radius along which the joint moves, as well

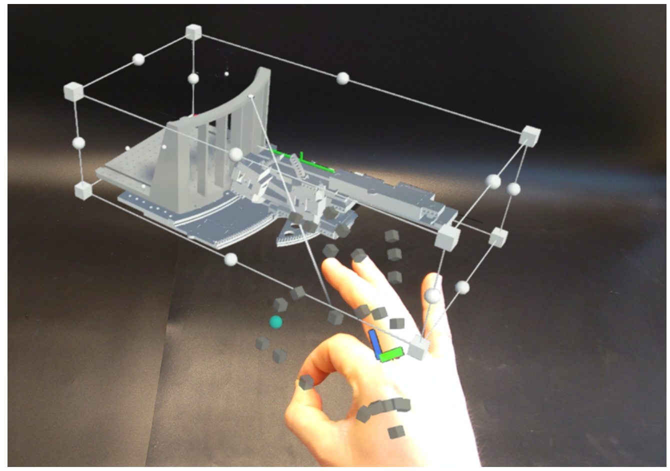


Fig. 32: The holographic scene surrounding by the bounding box, while the operator scales the scene. As the whole scene is scaled, relations between parts of the system remain unchanged.

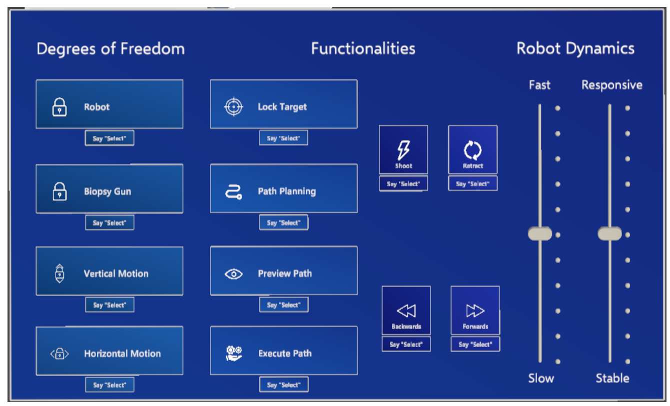


Fig. 33: The control panel, letting the operator lock or unlock certain DoFs of the robot, e.g. to only allow for movement of the biopsy gun for insertion of the needle. The modes of the robot can also be controlled, such as locking the target, path planning, previewing the path, and executing it. The biopsy gun can be fired, as well as retracted and the robot dynamics can be changed.

as the gear ratio of the motor. First, the distance in degree per step is calculated by the conversion formulas of Equation 13, 14. Afterward, the limits of the joint angles can be converted into "step-space" (Eq. 3). Finally, the current joint angle can be converted into the step position of the motor (Eq. 4).

The conversion from angle to step amount depending on the step size for motors with a 13:10 or 17:10 gear ratio:

$$d_{\text{step13:10}} = 180/\pi * 0.462\text{mm}/r_{\text{joint}} \tag{13}$$

$$d_{\text{step17:10}} = 180/\pi * 0.604\text{mm}/r_{\text{joint}} \tag{14}$$

The fixed constant in the conversion is the step size in mm, depending on the parameters of the motor. The pinion in the motor has 10 teeth and a gear modulus of 1, the steps per revolution are four times the amount of teeth of the outside gear. This results in step sizes in mm derived by Equations 15, 16.

$$l_{\text{step13:10}} = \frac{\pi}{4}\frac{10}{13} = 0.604\mathrm{mm} \tag{15}$$

$$l_{\text{step17:10}} = \frac{\pi}{4}\frac{10}{17} = 0.462\mathrm{mm} \tag{16}$$

Adjustable PD Control

Since the estimation of the hand position and orientation of the outside cameras of the HMD is susceptible to frame-by-frame variance, using this estimation directly to control the end-effector would introduce noise. A PD controller was implemented that defines the dynamics, with which the end-effector follows the input of the operator. This stabilizes the noise introduced by the estimation errors of the cameras. Additionally, the movement of the end-effector is scaled down w.r.t. to the input from the operator, such that larger movements of the hand of the operator result in smaller movements of the robot. Both the parameters of the PD controller and the scaling of movements were chosen arbitrarily through the subjective preference of the developer. The operator can adjust these values to their preference via the control panel (Fig. 33). Two sliders, one for the p-gain of the PD controller, and the other for the scale, can be adjusted to change the robot dynamics during the runtime of the application.

Adjustable Intensities of the MRI screen

The parameters of the MRI machine used in the image acquisition of the MRI data influence the resulting pixel values w.r.t. their intensities. This can cause a low contrast in the image, dependent on how the intensity values are mapped to the color depicted on the screen. Since a high-contrast image is crucial for the identification of potential lesions, being able to adjust the color values is a necessity. Sliders attached to the MRI screen facilitate this functionality (Fig. 34).

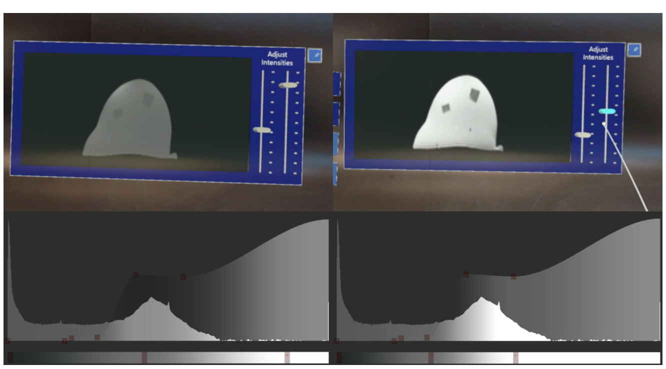


Fig. 34: The MRI screen before and after adjusting the transfer function defining the color intensities in the image, increasing the visibility of the target tissue.

During runtime, the operator can use these sliders to adjust the mapping of pixel intensity - which is dependent on the acquisition - to color values. This is done by assembling the pixel intensities as a histogram and applying a color gradient to it (Fig. 34). One slider moves a dark color along the length of the histogram, the other moves a lighter color.

APPENDIX D
FUTURE WORK

Real-time feedback

The current control paradigm is based on the initial diagnostic scan that is converted to a 3D mesh and imported to the holographic scene. Based on this calibration, feed-forward control is used to steer the end-effector to the desired position. From the 3D mesh and the MRI images alone, the operator only knows the current position of the end-effector relative to the initial scan. If the tissue deforms or moves during the procedure, there is no way of knowing until a control scan is conducted. The MRI scanner is not capable of imaging the entire field of interest in a short amount of time, but single slices can be taken with a relatively high frequency of up to 30 Hz [8]. This could be used to take control scans along the needle direction during the insertion to have live feedback on its trajectory. Some manufacturers of MRI scanning machines provide software/APIs, with which the control of the MRI machine through the AR application would be possible.

Since the software for the third-party control of the MRI machine was not available at the time of writing this paper, a demonstration based on the initial MRI scan was created. Two slicing planes are fixed to the surrogate along the needle trajectory, one in the vertical direction and the other in the horizontal direction. The corresponding MRI images, along with an indicator for the needle trajectory in the plane, are presented in a separate window (Fig. 35).

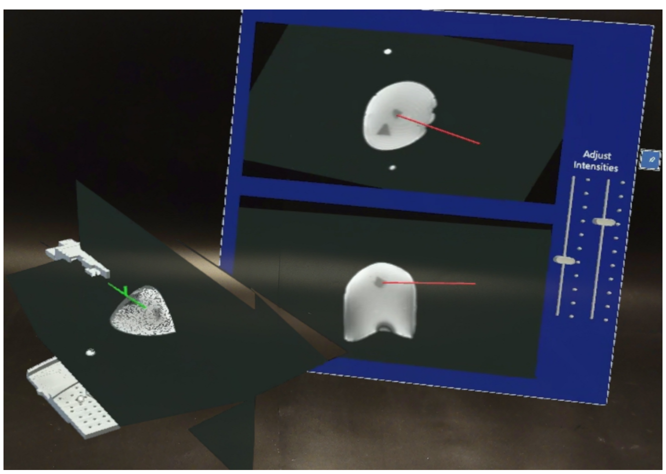


Fig. 35: Screenshot of the slicing planes along the needle and the corresponding MRI images including the needle trajectory.

Before and during the insertion of the needle, the operator can assess the orientation of the needle relative to the target lesion by observing its trajectory on the screens. In a subsequent implementation utilizing the MRI scanner, the positions of the slicing planes will be transformed into the MRI coordinate frame, and scans corresponding to those slices will be acquired and presented in AR. This approach will reveal potential deformations of the breast and any associated targeting errors, providing the operator with the opportunity to make necessary corrections.

Taking this approach further, the real-time update on the slices along the needle trajectory could be utilized to recalculate the 3D mesh in the holographic representation of the acquired MRI images.

Robot Position Feedback

Due to the circumstances in the MRI room, the physical robot controller, providing the pressurized air to the motors, must stay in the control room. Long tubes gap this distance, causing a drop-off in pressure due to the increased volume (Eq. 17), which makes a frequency reduction in order for the motors not to skip steps necessary.

$$V = r^2 \pi l \tag{17}$$

The resulting lower movement speed of the robot causes an inaccurate depiction of the holographic surrogate representing the physical robot. Currently, the operator relies on auditory feedback from the controller to know if the robot reached the desired position. Multiple other ways to give feedback w.r.t. the position of the robot could be used. Firstly, the surrogate could be slowed down to match the step frequency of the physical robot. This would most likely result in an unsatisfactory user experience and impede the use of the surrogate to facilitate the control. Secondly, a phantom, similar to the one used for path planning, could be used to represent the physical robot position. Other, simpler approaches, could be visual cues, such as progress bars or signal lights turning on once the desired position is reached.

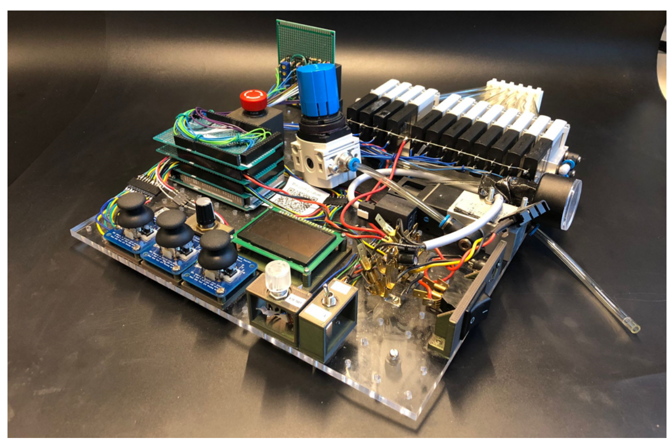
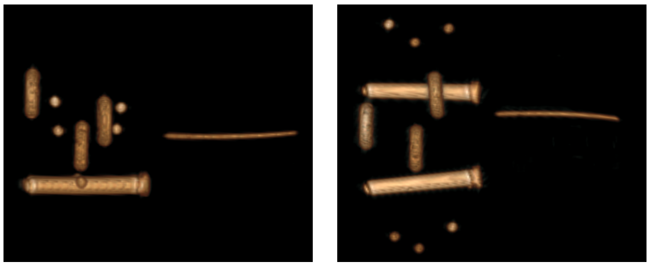


Fig. 36: Physical Robot controller.

Improved Calibration

The targeting accuracy of the system is to a certain degree dependent on the initial calibration accuracy. To fully harvest the potential of the system, efforts should be made to improve the calibration accuracy. The orientation of the markers used for the calibration potentially leaves room for slight rotational errors, that could be addressed by a different marker placement. Moving some markers from the side of the base of the robot to it's front and back, would increase their leverage w.r.t. the rotation, potentially decreasing the rotational error. Similarly, increasing the number of markers, or using markers even better detectable by the MRI machine, such as wireless coils, could improve the calibration accuracy [9], [10], [11].

Dependent on the resolution and slice orientation, the 3D reconstructed mesh used for targeting the lesions is morphing, resulting in an inaccurate representation of the actual tissue of the patient (Fig. 14).



(a) The 3D reconstruction of the coronal scan seen from the sagittal view.

(b) The 3D reconstruction of the sagittal scan seen from the coronal view.

Fig. 37: The 3D reconstruction from the MRI scan showing the view perpendicular to the scan direction. The line on the right side of each image is a petrolatum-filled mock needle, depicted as severely bent, although perfectly straight in reality.

This is currently circumvented by utilizing multiple scans with different orientations, namely coronal and sagittal scans. Since this, in its current implementation, slightly impedes the workflow of the system, methods to correct this warping are desirable. One method to counteract the warping of the 3D reconstruction was introduced by Groenhuis et al. using a fifth-order correction polynomial, to analytically correct the error evaluated by a 3D calibration cube [32]. After further investigation toward the end of this work, it became evident, that the software of the MRI machine includes such a utility. Further experiments should be conducted to re-evaluate the performance of the system based on the corrected 3D reconstructions of the MRI images.

Improving Accuracy

Another cause of errors is the nature of the stepper motors used by the robot. The target that the robot approaches is defined in a quasi-continuous space, the stepper motors however force the robot to move in a discrete space. Dependent on the radius along which the respective link of the robot moves, as well as the gear ratio of the motor, one step of the motor can be translated to the amount of degrees the link moves (Eq. 4). The end-effector operating at a distance to those joints, results in leverage of this displacement dependent on its extension w.r.t. the needle tip. In the most extreme case of the joint most distant to the needle tip, one step of the motor corresponds to a change in angle of $0.577°$. When the end-effector is fully extended, the lever to the needle tip is approximately 300 mm, resulting in a displacement at the needle tip of $3\mathrm{mm}$. In practice, the effect is less detrimental than that, as it is compensated by the second joint in the same plane, although an error remains.

Taking the constraints of the stepper motors into account when targeting could increase the accuracy of the robot. An optimization algorithm could find the robot configuration, that respects the motor step sizes, and minimizes the Euclidean distance error to the target. This optimization could be thresholded and combined with a cost function w.r.t the initially desired position. When in the auto-targeting mode, the robot could be guided to the desired position, after which it automatically finds a nearby minimum and approaches it.

Integrated User Guide

To familiarize the user with the system, which capabilities might not be evident from the UI alone, an introduction or explanation is needed. Instead of using trained staff to give this introduction, the potential of the spatial computing of the HMD could be utilized. When entering the application for the first time, the user could be guided through an AR-based tutorial in which the functions of the system are presented and explained. The user follows each step to apply and learn each functionality. Additionally, voice commands could be used to revisit these tutorials for specific functionalities. These tutorials could even be implemented in a way that does not impede the workflow of the system, but rather co-exist, such that the user can inquire about them at any given time.

Improving the Software Architecture

Although the code was refactored multiple times during the development, new learnings were gathered that could not be implemented in time. Due to the lack of these learnings in the starting phase of the development, sub-optimal software structures lay the foundation of the project. Before taking the system into a more developed stage, rebuilding the project should be considered.

Intuitive/Self Adjusting Voice Commands

Currently, to utilize voice commands for the control of functionalities of the system, the user has to speak the exact sentences hard-coded into the program. Oftentimes a command could be described with similar words, effectively carrying the same meaning. In the example of the user wanting to use the path planning mode of the system, both "Activate Path Planning" and "Enable Path Planning" convey the same meaning, whereas only one will activate the path planning mode. Theoretically, all possibilities could be hard-coded into the system to facilitate an activation of certain functionalities through different voice commands, which would however be inaccurate and most likely prone to edge cases that are still missing. The recent developments in AI, more specifically the availability of large language models (LLM), provide an elegant solution for such a problem. A prompt could be used to give the LLM the target outputs of the voice commands, it can then be instructed to interpret an arbitrary command given by the user and map it to one of the target outputs. To prevent the model from constantly interpreting everything the user says, a cue can be given to signal the intent of giving a command. This could be in the form of a name, similar to companies giving their assistive AI names that are called before a question is asked (e.g. Siri, Cortana etc.).

REFERENCES

[1] H. Temiz, B. Gökberk and L. Akarun, "Multi-view Reconstruction of 3D Human Pose with Procrustes Analysis," 2019 Ninth International Conference on Image Processing Theory, Tools and Applications (IPTA), Istanbul, Turkey, 2019, pp. 1-5, doi: 10.1109/IPTA.2019.8936071.

[2] A. R. Anwary, H. Yu and M. Vassallo, "Gait Evaluation Using Procrustes and Euclidean Distance Matrix Analysis," in IEEE Journal of Biomedical and Health Informatics, vol. 23, no. 5, pp. 2021-2029, Sept. 2019, doi: 10.1109/JBHI.2018.2875812.

[3] K. F. Zaidi and M. Harris-Love, "A Novel Procrustes Analysis Method to Quantify Multi-Joint Coordination of the Upper Extremity after Stroke," 2023 45th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), Sydney, Australia, 2023, pp. 1-4, doi: 10.1109/EMBC40787.2023.10341023.

[4] O.G Salem, Connected Component Labeling Algorithm, (2014), Code Project Article, https://www.codeproject.com/Articles/336915/Connected-Component-Labeling-Algorithm

[5] Gamma, E., Helm, R., Johnson, R.,, Vlissides, J. M. (1994). Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley Professional. ISBN: 0201633612

[6] D Zhart, Refactoring Guru, accessed 2023, https://refactoring.guru/

[7] M. Lavik, Unity Volume Rendering, (2019), GitHub repository, https://github.com/mlavik1/UnityVolumeRendering

[8] Hirsch, F. W., Frahm, J., Sorge, I., Roth, C., Voit, D., & Gräfe, D. (2021). Real-time magnetic resonance imaging in pediatric radiology — new approach to movement and moving children. In Pediatric Radiology (Vol. 51, Issue 5, pp. 840–846). Springer Science and Business Media LLC. https://doi.org/10.1007/s00247-020-04828-5

[9] C. -L. Cheung, J. D. -L. Ho, V. Vardhanabhuti, H. -C. Chang and K. -W. Kwok, "Design and Fabrication of Wireless Multilayer Tracking Marker for Intraoperative MRI-Guided Interventions," in IEEE/ASME Transactions on Mechatronics, vol. 25, no. 2, pp. 1016-1025, April 2020, doi: 10.1109/TMECH.2020.2970990.

[10] C. -L. Cheung et al., "Omnidirectional Monolithic Marker for Intra-Operative MR-Based Positional Sensing in Closed MRI," in IEEE Transactions on Medical Imaging, vol. 43, no. 1, pp. 439-448, Jan. 2024, doi: 10.1109/TMI.2023.3309967.

[11] C. D. Jordan et al., "Wireless Resonant Circuits Printed Using Aerosol Jet Deposition for MRI Catheter Tracking," in IEEE Transactions on Biomedical Engineering, vol. 67, no. 3, pp. 876-882, March 2020, doi: 10.1109/TBME.2019.2922879.

Appendix E
General Remarks

Use of AI Tools

The use of the recently rapidly developing generative AI tools provides the opportunity to increase productivity when used correctly. Trivial parts of a program, commonly referred to as ”boilerplate”, can easily be generated by these tools, accelerating the process from the idea to a functioning product. At the time of writing this, these tools can’t yet generate more complex programs, such as the software presented in this work. Especially, when using a programming language and a piece of software for the first time, like was the case in this work for C# and Unity, tools like ChatGPT can be tremendously helpful with answering quick questions. During the development of the software, these tools were used to provide answers to syntax questions w.r.t the C# programming language and the inner workings of the Unity game engine. Generative AI also provides help with writing, by finding synonyms and alternate formulations of sentences. Parts of this work were pre-written and fed into such a tool to provide alternative words and slightly altered sentences, accelerating the otherwise lengthy process, when doing the same manually.

Acknowledgements

First and foremost I want to thank Dr. Kenan Niu and Dr. Vincent Groenhuis for their amazing supervision. I appreciate your efforts and constant availability a lot. I enjoyed every day working on this thesis and hope our work can continue in one way or the other. I also want to thank Dr. Kenan Niu, Dr. Vincent Groenhuis, and Prof. Dr. Stefano Stramigioli for the many opportunities they created for me during the period of this thesis. I tried my best to use each and every one of these opportunities and won’t take them for granted. I want to thank Dr. Wyger Brink for his significant contribution to this work, through his expertise during the experiments in the MRI room and the many hours selflessly spent with the control of the MRI scanner.

My biggest and most heartfelt gratitude goes towards my parents, who created the opportunity for me to find my passion by supporting me unconditionally. I am very aware of the privilege I received from having you in my life and I will forever be grateful for the worry-free life you gave me. I want to thank my brother for being a great role model and giving me the confidence to approach any challenge with determination. Lastly, I want to thank Cèlia Vicens, without whom I wouldn’t have been able to catch up on the knowledge I lacked at the beginning of my studies. Thank you for answering my many engineering questions and being an amazing friend.

An Augmented-Reality based digital twin for controlling an MR safe robot in breast biopsy

Lennard Marx[1], Vincent Groenhuis[1], Stefano Stramigioli[1] and Kenan Niu[1]

***Abstract*— Magnetic resonance (MR) guided breast biopsies can be demanding and time-consuming procedures for both patients and surgeons. Inaccurate needle insertion can lead to false negatives, resulting in faulty diagnoses with serious consequences. Incorporating robots into these procedures has the potential to improve accuracy, reduce patient strain, and decrease MRI time and costs. For the surgeons to remain in control and make adjustments, a control interface is a necessity. In this study, an augmented reality (AR)-based control method for surgeons is proposed by offering a new means of interacting with robots with enhanced intuition. A virtual surrogate of the MR safe robot that can be controlled through gestures and voice commands in the AR scene was developed. This virtual surrogate can be operated outside of the bore of MRI scanner. Moreover, the proposed system enables collision avoidance and supports semi-autonomous behavior for targeting lesions and performing biopsies. The system was evaluated by comparing two different AR based control strategies to manual control using physical joysticks. The gesture based control method exhibited an accuracy of 2.44 $\pm$ 1.32 mm and a precision of 1.64 $\pm$ 1.00 mm. In the auto-targeting mode, where the subject moved the locked target, an accuracy of 2.72 $\pm$ 1.58 mm and a precision of 1.84 $\pm$ 1.39 mm was achieved. These findings highlight the efficacy of holographic surrogate control in enhancing the efficiency, precision and user experience of robot tele-operation. Moving forward, it will facilitate the application of MR safe robots in breast biopsy.**

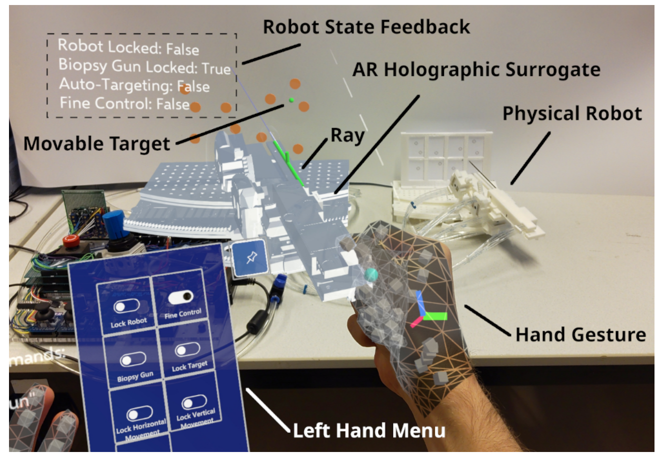


Fig. 1: The demonstration of AR-based digital twin for controlling Sumram7 robot. The Sunram7 robot imitates the scaled-up holographic surrogate. A ray is controlled via the hand forming a fist gesture, to which the holographic surrogate aligns its end-effector. The green dot indicates the movable target used in the auto-targeting mode. The left hand menu is to adjust the functionalities of the robot is visible, when the operator opens their palm and faces it towards the HMD.

I. INTRODUCTION

Breast cancer accounts for nearly one-third of cancer cases in women worldwide [1]. With a gradually increasing prevalence in breast cancer and a mortality rate of 10%, which is considerably higher for individuals with limited access to preventive care, an early diagnosis becomes increasingly important. There is an urgent need for the development of novel diagnostic approaches. The primary methods for diagnosing breast cancer are mammography and ultrasound. However, in cases involving a heightened risk of breast cancer, such as a high familial predisposition, mammography, even when combined with ultrasound, may prove insufficient for early detection [2]. The utilization of magnetic resonance imaging (MRI) has the potential to significantly enhance diagnostic sensitivity and enable earlier detection [2]. Nonetheless, the MRI-guided breast biopsy procedure can be time-consuming and physically intensive for patients [3]. To ensure precise needle placement during the biopsy, multiple interim MRI scans are conducted. In between these scans, the patient must be removed from the MRI bore to readjust the needle position. MR safe robots, such as the Sunram7 (Fig. 4a), offer the potential to expedite this process by enabling remote adjustments from a control room [4].

Augmented reality (AR) technology can be employed to visualize MRI images as three-dimensional (3D) objects through head-mounted displays (HMD) and facilitate intuitive path planning via projecting 3D Holographs of segmented anatomical structures [5]. Concurrently, AR can be used to interact with and teleoperate robots using gestures, voice commands, and holographic interfaces [6], [7]. An AR interface for teleoperating and potentially automating MRI-compatible robots like the Sunram7 could significantly simplify the breast biopsy procedure, making it less of a burden for patients and more intuitive for surgeons. Moreover, it could reduce the time duration and, consequently, the cost of the procedure. This, in turn, could enable MRI as a cost-effective diagnostic method, potentially increasing its utilization for early-stage diagnosis [8].

A. State-of-the-art

Currently, augmented reality (AR) is predominantly used for visualization purposes and facilitating human-robot collaboration in various applications [6]. However, there exists

[1]Robotics and Mechatronics, University of Twente, Enschede, AE, The Netherlands l.marx@student.utwente.nl (L.M.), v.groenhuis@utwente.nl (V.G.), s.stramigioli@utwente.nl (S.S.), k.niu@utwente.nl (K.N.)

Manuscript 2453 submitted to 2024 IEEE International Conference on Robotics and Automation (ICRA). Received September 15, 2023.

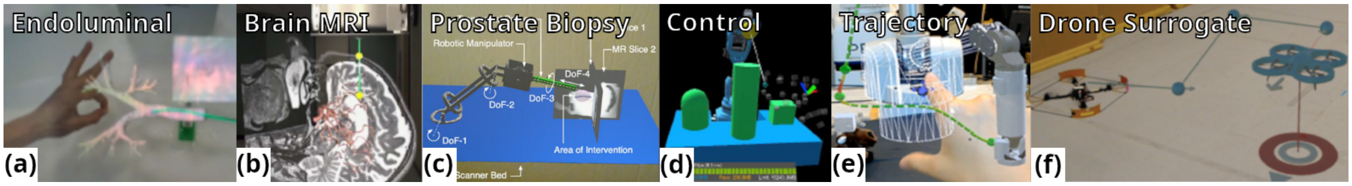


Fig. 2: Various AR-based interactive applications exerted from the state-of-the-art referenced here (Sec. I-A). (a): Lin et al. [9], (b): Morales Mojica et al. [10], (c): Velazco-Garcia et al. [11], (d): Xue et al. [12], (e): Quintero et al. [14], (f): Walker et al. [15].

limited research on the use of AR for robot control, particularly in the context of medical applications. For instance, Lin et al. developed a teleoperated robot for endoluminal interventions, which is controlled through gestures using a HMD [9]. A hologram of the trachea and the endoscope is presented in front of the user, allowing the user to manipulate the endoscope's position within the trachea through hand gestures. Different hand gestures were used for different degrees of freedom (DoF), making a simulateous control of multiple DoFs impossible. Similarly, Morales Mojica et al. used AR for image-guided control of interventional manipulators [10]. They visualized MRI images as holograms in front of the user, offering the ability to manipulate and display specific slices of the MRI scan. The AR interface was simultaneously used to control the manipulator, as well as to regulate the incision depth. This was done by defining a target point in a 2D plane and an incision point in the same plane, direct control of the robot was not possible. In a related study, Velazco-Garcia et al. conducted a comparative analysis of different input methods for planning MR-guided prostate biopsies [11]. They evaluated a holographic interface, comparing it with Gamepad and Mouse/Keyboard input methods. In this setup, the robot, along with the organs and lesions, were rendered as meshes in front of the user, enabling visualization of the robot's workspace. Notably, Velazco-Garcia et al. concluded that the AR input method yielded the least favorable results [11]. The sensitivity of the interactions with the holograms made it challenging to make precise adjustments along one dimension without inadvertently affecting another, ultimately deeming this method impractical.

While this paper primarily centers on the medical application of AR, it is essential to provide a comprehensive overview of the state-of-the-art focusing on AR for robot control, including non-medical applications. Xue et al. introduced an AR-based robot control interface that allows users to manipulate the virtual end-effector to a desired position [12]. The inverse kinematics (IK) are resolved by the FABRIK method [13]. Additionally, they implemented collision detection to ensure a safe interaction with the environment. Quintero et al. programmed robots by defining trajectories within an AR environment [14]. These trajectories could be established by placing AR waypoints in 3D space, automatically generating a corresponding trajectory. Moreover, they introduced the capability to simulate tasks using a virtual surrogate before executing them with the physical robot. An impedance control mode was enabled by planning trajectories along surfaces along which the robot applied a constant force. Issues were reported concerning the stability of hologram locations when operators moved while wearing the HMD. Walker et al. employed two distinct AR-based control methods to manage a drone via a holographic surrogate: real-time PID control and waypoint control [15]. Operators could either directly move the virtual surrogate, causing the drone to follow, or lock the drone's position and create waypoints by manipulating the virtual surrogate.

In this paper, the aim is to introduce an AR-based approach for intuitive interaction and tele-operation of an MR-safe robot. The proposed method uses a HMD to present a holographic surrogate of the robot. The surrogate robot and its actual physical presentation (i.e. real robot) are linked as a digital twin to reproduce the same robotic executions. By utilizing gestures and voice commands, the operator gains the immersive ability to engage with the holographic surrogate, enabling precise control of the robot from the operating room. Concurrently, common challenges associated with AR-based control, such as the heightened sensitivity of the hologram, which can lead to reduced targeting accuracy are addressed. While this work focuses on the medical application, it is worth noting that the proposed approach and its implementation hold the potential for adaptation to a wide range of robotic manipulators.

II. METHODS

A. *Overview of the Framework*

Figure 3 provides an overview of the intended use case of the system. The process begins with an MRI scan to generate the virtual scene and calibrate the virtual robot's position to match the physical robot's position. The operator in the control room wears the Hololens 2 HMD, which renders the hologram representing the scene and recognizes their input through gestures and voice commands. This input is used to calculate the desired joint positions through the IK and control the robot with the intended dynamics via the controller. Visual feedback from the holographic scene is provided to the operator to make necessary adjustments. Subsequently, the HMD sends the intended joint positions via WIFI, either through a computer or directly to the micro-controller of the robot. Finally, the robot executes the commands in the MR room.

Manuscript 2453 submitted to 2024 IEEE International Conference on Robotics and Automation (ICRA). Received September 15, 2023.

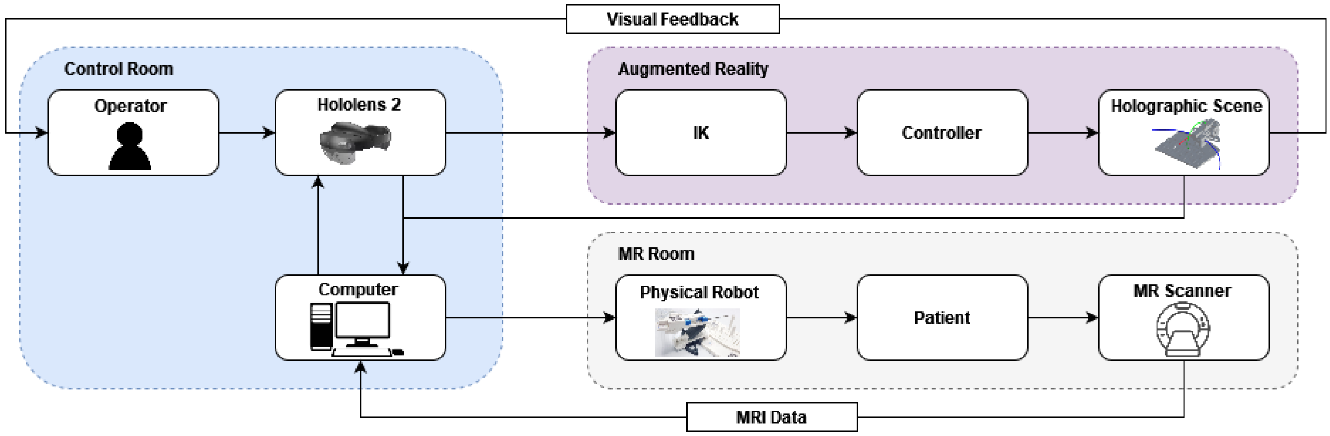
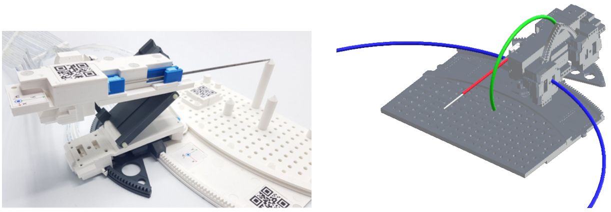


Fig. 3: The schematic overview of different components in the system in the intended use case of MR-guided biopsy.



(a) The Sunram7 robot. (b) The virtual surrogate.

Fig. 4: The Sunram7 robot and the virtual surrogate. The blue and green circles on the surrogate represent the radii on which the primary joints of the robot move.

B. Sunram7 robot configuration

The Sunram7 is a five-degree-of-freedom MRI-safe robot designed for breast biopsy procedures [4]. To ensure compatibility within an operational MRI environment, the robot's construction relies entirely on 3D-printed plastic components. The Sunram7 utilizes pneumatic stepper motors for its actuation, which are also manufactured using rapid prototyping techniques [16]. Its five degrees of freedom consist of four revolute joints and one prismatic joint. Specifically, two revolute joints facilitate horizontal movements of the robot, while the other two enable vertical motion. The prismatic joint controls the movement of the end-effector, responsible for guiding the biopsy needle along its designated trajectory.

C. Augmented reality-based robotic control framework

For software development, the Unity game engine (Unity Technologies, San Francisco, U.S.) was employed in conjunction with the Mixed-Reality-Toolkit (MRTK) (Microsoft, Redmond, Washington, U.S.). The MRTK is an open-source software development kit (SDK) designed for creating mixed reality (MR) and AR applications. The application was deployed on the Hololens 2 MR and AR HMD (Microsoft, Redmond, Washington, U.S.).

To generate the holographic representation of the robot, the same CAD (computer-aided design) files used for 3D printing the Sunram7 were imported into Unity and configured to replicate the robot's physical structure. A custom script was developed to solve the IK, allowing the robot's end-effector to align itself with a designated ray, thereby defining the desired position and orientation. Due to the robot's design, the IK problem can be solved in two dimensions independently. Figure 4b illustrates the two main circles on which the robot moves, as well as the ray. In each dimension, two angles, denoted as θ_1 and θ_2, need to be calculated. This is achieved by first determining the intersection point of the ray with the circle and then deriving the angle offset of that point from the baseline (Fig. 5). First, the perpendicular distance from the ray to the center of the circle is calculated to verify if it is smaller than the circle's radius. If it is not, no intersection can be found, and the ray is outside the robot's workspace (Eq. 1). Subsequently, the intersection point p_1 of the ray with the circle can be computed, enabling the straightforward derivation of angles θ_1 and θ_2 (Eq. 2).

$$y = s - (r_0 + \vec{r_d}((s - r_0) \cdot \vec{r_d}) \tag{1}$$

$$p_{1,\,2} = r_0 + \vec{r_d}(((s - r_0) \cdot \vec{r_d}) \pm \sqrt{r^2 - y^2}) \tag{2}$$

To make the operator's interaction with the robot more intuitive, the ray is controlled through hand gestures and

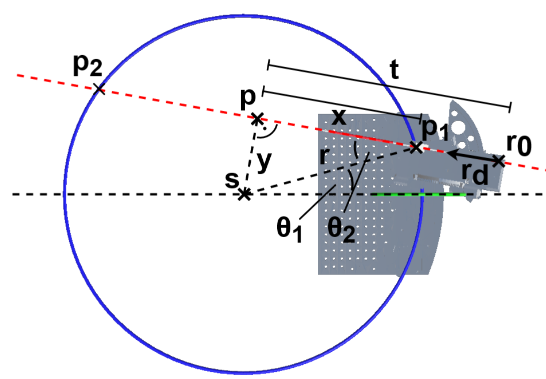


Fig. 5: The diagram visualizes the variables used in the IK calculations.

voice commands. Hand motions are considered commands only when the user closes their hand, effectively implementing a "reconfiguration clutch," a feature commonly found in many commercial surgical robots for hand reconfiguration purposes, typically activated using a foot pedal. The MRTK is employed to detect the operator's hands, from which gestures can be derived. Robot motion is defined by a proportional derivative (PD) controller that takes into account the error between the robot's end-effector and the ray, effectively filtering out high-frequency noise from the estimation of hand position. To further enhance the stability of robot control, the hand's motion is down-scaled by a factor of 4, meaning that 100 mm of movement by the operator translates to 25 mm of movement of the robot. Alternatively, the robot can automatically align itself with a predefined target upon receiving the command. This target can be moved in 3D space by the operator using the default interaction method built into the HMD, which consists of a beam extending from the operator's hand, allowing objects to be manipulated with a pinching gesture (Fig 6(b)). The system also supports voice commands, which are mapped to specific robot functionalities. As an alternative input method to voice commands, all instructions can be provided through a menu that appears when the operator opens their left hand and positions their palm toward the HMD (Fig. 1). The current joint position is then forwarded through serial commands to the physical robot. Given that the robot uses stepper motors, the joint positions must be converted to step positions of the motors. First, the change in angle per step is calculated, which depends on the gear ratio of the motor and the radius on which the joint moves. Equations 3 and 4 demonstrate the conversion from angle rate to step rate.

$$d_{\text{step13:10}} = 180/\pi \cdot 0.462/r_{\text{joint}} \tag{3}$$

$$d_{\text{step17:10}} = 180/\pi \cdot 0.604/r_{\text{joint}} \tag{4}$$

The workspace limits of the robot can then be calculated by equation 5.

$$p_{\min} = \frac{\theta_{\min}}{d_{\text{step}}}, p_{\max} = \frac{\theta_{\max}}{d_{\text{step}}} \tag{5}$$

Finally, the current joint angles are mapped to the step position with equation 6.

$$p_{\text{steps}} = p_{\min} + (\frac{\theta - \theta_{\min}}{\theta_{\max} - \theta_{\min}} \cdot (p_{\max} - p_{\min})) \tag{6}$$

D. Interactive control strategies

The integration of these features gives rise to two distinct control strategies. The first strategy involves controlling the robot's end-effector by initiating movement with a fist gesture and positioning the robot by manipulating the ray. Additionally, specific DoFs can be locked to facilitate minor adjustments along one dimension without inadvertently impacting others. The second control method leverages the robot's auto-targeting mode. In this mode, the operator can manipulate and move the target to a desired landmark, and subsequently, the robot's end-effector will follow this movement. Position control of the robot is still possible in this control mode, as the robot will maintain the target in aligned with its end-effector.

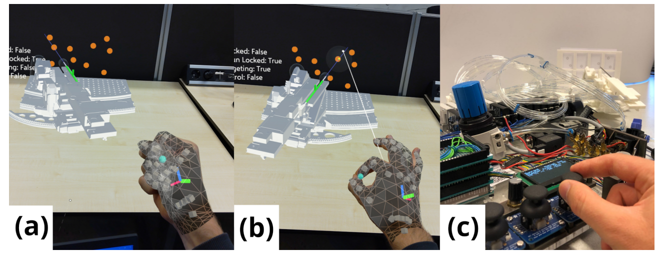


Fig. 6: The three different control methods from left to right: (a) Gesture control, (b) auto-targeting mode, and (c) joystick control.

E. Experimental evaluation

To evaluate the effectiveness of the proposed methods, a pseudo-randomized study involving ten subjects was conducted. A sheet of paper containing ten designated targets was placed in a holder in front of the robot. Subjects were instructed to puncture the paper at the target locations using the biopsy needle. Both the total time required to puncture all ten targets and the distance from the penetration point to the center of each target were measured. Three distinct control methods were tested and subsequently compared in this manner. The first two methods correspond to those previously described (Section II-D): directly moving the end-effector and moving the auto-locked target. The third method involved controlling the joint angles via physical joysticks (Fig. 6(c)). To minimize potential bias due to training effects, the order of the trials was alternated for each subject. Five subjects started with the AR-based methods followed by the physical control method, while the other five subjects started the other way around. Subsequently, the accuracy, precision, and task completion times achieved with the three different control methods were compared.

III. RESULTS

Ten subjects with limited or no prior experience with AR participated in the experiments. Figure 7 provides an overview of all trials, with rows representing the different methods and columns representing the different subjects. In a few trials, noticeable offsets from the center of the target can be observed (e.g., Subject 7). Figure 8 displays the accuracy and precision of the incision point relative to the target center for all trials, along with the time spent on each trial. Table I summarizes the results for all three methods. The leftmost comparison assesses the accuracy of the trials by analyzing the raw data collected (Fig. 8(a)). The average accuracy was 2.44 $\pm$ 1.32 mm for the gesture control method, 2.72 $\pm$ 1.58 mm for the auto-targeting method, and 3.71 $\pm$ 2.42 mm for the joystick control method. A pairwise analysis of variance (ANOVA) was conducted to identify significant differences between the methods. Both AR-based methods outperformed the joystick method significantly.

Manuscript 2453 submitted to 2024 IEEE International Conference on Robotics and Automation (ICRA). Received September 15, 2023.

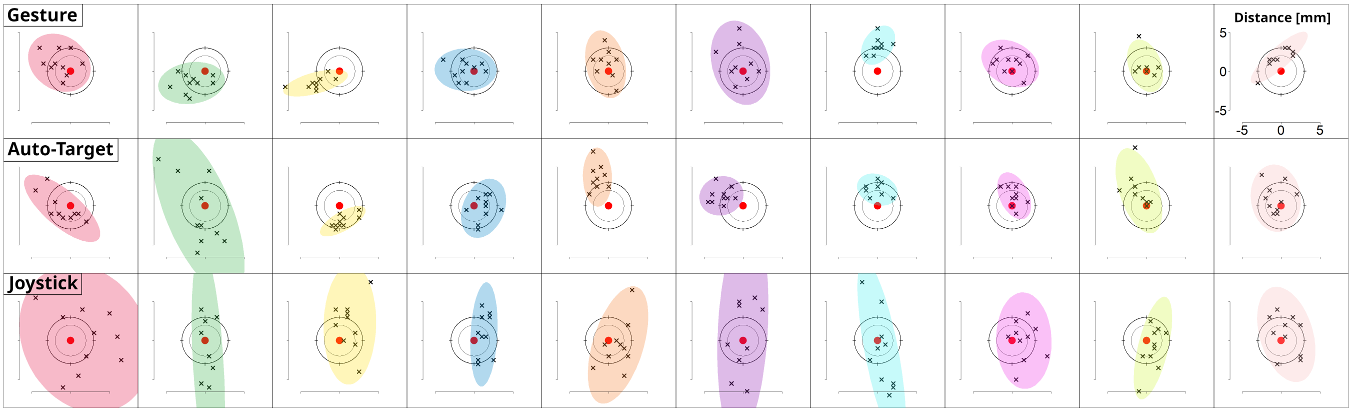


Fig. 7: The overview of all trials of ten subjects. The rows represent the three different methods used, the columns/colors represent the different subjects. The outer circle of the target has a radius of 3mm.

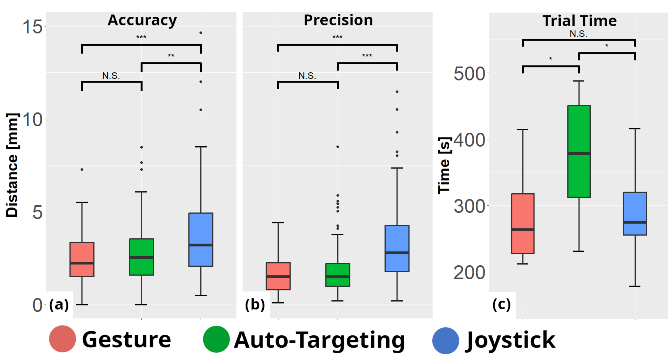


Fig. 8: Boxplots comparing the three different methods. (a) shows the raw data of the distance in mm to the target of all trials. (b) To show the precision, the inter-subject mean vectors where subtracted from each data point. (c) shows the trial times for each of the methods in seconds. The lines above the plots indicate comparisons via a pairwise ANOVA. The asterisk indicate the order of significance.

TABLE I: The results of the experiments for each of the methods used in the experiments.

Method	**Accuracy** [mm]	**Precision** [mm]	**Time** [s]
Gesture	2.44 ± 1.32	1.64 ± 1.00	279.1 ± 64.5
Auto-Target	2.72 ± 1.58	1.84 ± 1.39	377 ± 83.4
Joystick	3.71 ± 2.42	3.32 ± 2.30	290.4 ± 70.6

To assess the precision of the methods, offsets (e.g. caused by calibration errors or subject-specific biases) were mitigated by subtracting the inter-subject mean vector from the trials. The resulting average scores were 1.64 ± 1.00 mm, 1.84 ± 1.39 mm, and 3.32 ± 2.30 mm for gesture control, auto-targeting, and joystick control, respectively. The pairwise ANOVA revealed significant differences between the joystick control method and the AR-based methods, with the AR-based methods demonstrating significantly better performance.

Finally, the time it took each subject to complete a trial (puncturing 10 targets with the needle tip) was assessed. On average, subjects took 279.1 ± 64.54 s for gesture control, 377 ± 83.42 s for auto-targeting, and 290.4 ± 70.59 s for joystick control to complete the trials. A pairwise ANOVA analysis was conducted once more, revealing significant differences between the auto-targeting method and the others, with the auto-targeting method performing significantly worse in terms of time efficiency.

IV. DISCUSSION

In this paper, two AR-based methods for tele-operation of an MR-safe robot were developed to create an intuitive and precise method of controlling the robot and targeting lesions in 3D space during breast biopsy. Both the accuracy and precision of the methods (Fig. 8) exhibited a significant increase in both metrics compared to the manual joystick control method. While the subjects had visual feedback from the real targets when using the joystick control, they only had a virtual representation of the targets with the AR-based methods, which implies that real tele-operation would only be feasible with the proposed methods. Nonetheless, the joystick control method still performed significantly worse than the AR-based methods. Figure 7 reveals that many trials exhibited biases, which could have originated from various sources. In the AR-based methods, the calibration of the robot to the holographic representation is assumed to be a major source of the offset. With the joystick control method, there was a much greater spread of the distance to the target along the y-axis. This discrepancy could be attributed to the visual feedback being clearer in terms of the horizontal alignment of the biopsy needle to the target when sitting behind the robot. To mitigate these biases, the inter-subject mean vector was subtracted from the trials, centering the average of each trial on the target. This allowed for the evaluation of the precision of the different methods. With the inter-subject mean subtracted, the average precision was shown to be 1.64 ± 1.00 mm and 1.84 ± 1.39 mm for both AR-based methods, performing almost twice as well as the joystick method. These values could already be sufficient for performing biopsies, which are rarely performed on tumors smaller than 5 mm in diameter [17]. For comparison, other

Manuscript 2453 submitted to 2024 IEEE International Conference on Robotics and Automation (ICRA). Received September 15, 2023.

biopsy methods report a targeting error of 4.4 $\pm$ 2.9 mm as a good result, although those were evaluated in a real biopsy environment [18]. Arguably, these values can be further improved. The subjects had little to no experience and limited time to become accustomed to the interaction with the holograms. With more training and further improvement in the interfacing and control, the precision could be increased. Additionally, the precision is limited by the step size of the motor. With a gear ratio of 13:10, as used in the experiments, one step corresponds to an angle difference of $0.577°$ at the joint. At the joint that is furthest away from the needle tip, at distances of 200 mm and 300 mm when the needle is retracted and extended respectively, this translates to a theoretical displacement of approximately 2.014 mm and 3.021 mm at the needle tip. In practice, the other joint in that degree of freedom compensates for this displacement, but it is certainly a limiting factor. These results demonstrate that, especially with the potential for further precision improvement, the proposed methods effectively address the concern expressed in other literature that AR might be too sensitive for precise control [11]. Concerning the time taken by the subjects to complete the trials, there was no significant improvement from the gesture-based method to the joystick control method. Here, additional training and greater familiarity with the system could enhance the speed at which subjects can target with the biopsy needle. The auto-targeting method allows the robot to automatically track a predefined target, which can be manipulated by the operator. Even though grasping the target was made easier by a transparent halo that effectively expanded the graspable area manifold, this proved to be challenging for the subjects to adapt to and represented the most significant limitation of the auto-targeting method. Finding a more user-friendly method for moving the target could potentially improve both the speed and precision of this method. For instance, a method similar to the general control of the robot proposed here could be employed. One of the greatest benefits of using AR in the proposed methods was the scalability of the scene and its flexibility regarding positioning relative to the hologram. Subjects could scale up the size of the scene, making the virtual representation of the robot and the targets more than twice as large as their physical counterpart. Additionally, they could position the virtual surrogate favorably to clearly observe the trajectory of the biopsy needle and even adjust their positions to the hologram during the trials. Scaling and flexibility significantly improved precision, as small hand movements had a reduced impact. With the incorporation of the already scaled-down hand movements to the robot and the PD controller, precise control became feasible. One could argue that visualizing the end-effector without simulating the entire robot would suffice. However, this approach has the advantage of providing a better visualization of the robot's workspace constraints, making the restrictions on the end-effector's position more apparent to the user. Furthermore, potential collisions can be easily avoided in the software, as opposed to performing complex calculations on the robot's constraints in various joint configurations. Additionally, the current position of the robot can be visualized by the holographic surrogate, rather than only displaying the desired position of the end-effector.

Currently, the primary limitation of the proposed method is the calibration of the robot. Since the physical robot moves in steps, a flawed calibration can result in an offset of the end-effector position by up to 3 mm. Secondly, although this paper effectively addressed the issue of hologram instability, especially concerning hand position estimation by the HMD, remaining jittering causes slight inaccuracies. These unintended motions are challenging to filter out without making the entire system feel sluggish. Lastly, it is crucial that the real-world scene is accurately translated into the virtual world. This introduces another potential source of error. If the target is inaccurately represented in the hologram (e.g., with an offset from its actual location relative to the robot), the real target could potentially be missed, even though the virtual target is precisely hit.

To improve the system further, a calibration system should be implemented to eliminate the noticeable offset observed in some of the trials. Additionally, the interaction with the holograms can be refined and made more intuitive. However, the primary focus will be on introducing additional features specific to the intended application - breast biopsy. This will involve representing the patient's anatomical structures along with the target lesion as a hologram directly constructed from MRI images. One potential solution could be to insert a plane into the 3D mesh representing the breast and display the corresponding MRI slice on which the lesion can be marked. This would create a 3D spatial target that can be automatically tracked by the system. The operator could then maneuver the robot to the desired incision point, avoiding specific anatomical structures during the biopsy.

V. CONCLUSIONS

An AR-based digital twin control scheme for surgeons to interact with an MR-safe robot has been developed. This scheme involves creating a holographic surrogate of the robot and the biopsy target, enabling intuitive interaction through gestures and voice commands for the operator. Experiments were conducted with ten subjects to assess the performance of two distinct AR-based control methods and compare them to manual control via joysticks. The results demonstrated a significant advantage in terms of precision and accuracy compared to the manual control method. Achieving a precision of 1.64 $\pm$ 1.00 mm, it is feasible to target tumors with a diameter of $\sim$ 5 mm while tele-operating the robot from the control room. Moving forward, it is essential to integrate the system into a real MR environment to close the loop between the imaging process and the biopsy procedure. This integration could involve visualizing the anatomical structures of the patient to target the lesions accurately and providing visual feedback during the incision, thereby enhancing the overall system functionality.

Manuscript 2453 submitted to 2024 IEEE International Conference on Robotics and Automation (ICRA). Received September 15, 2023.

References

[1] Siegel RL, Miller KD, Fuchs HE, Jemal A. Cancer statistics, 2022. CA Cancer J Clin. 2022 Jan;72(1):7-33. doi: 10.3322/caac.21708. Epub 2022 Jan 12. PMID: 35020204.

[2] Kuhl, C., Schrading, S., Leutner, C., Morakkabati-Spitz, N., Wardelmann, E., Fimmers, R., Kuhn, W., Schild, H. (2005). Mammography, Breast Ultrasound, and Magnetic Resonance Imaging for Surveillance of Women at High Familial Risk for Breast Cancer. Journal of clinical oncology : official journal of the American Society of Clinical Oncology. 23. 8469-76. 10.1200/JCO.2004.00.4960.

[3] Papalouka V, Kilburn-Toppin F, Gaskarth M, Gilbert F. MRI-guided breast biopsy: a review of technique, indications, and radiological-pathological correlations. Clin Radiol. 2018 Oct;73(10):908.e17-908.e25. doi: 10.1016/j.crad.2018.05.029. Epub 2018 Jul 2. PMID: 30041954.

[4] H. Ranjan, M. van Hilten, V. Groenhuis, J. Verde, A. Garcia, S. Perretta, J. Veltman, F.J. Siepel, S. Stramigioli, "Sunram 7: An MR Safe Robotic System for Breast Biopsy". IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Oct 2023, pp. In press.

[5] Morales Mojica CM, Velazco-Garcia JD, Pappas EP, Birbilis TA, Becker A, Leiss EL, Webb A, Seimenis I, Tsekos NV. A Holographic Augmented Reality Interface for Visualizing of MRI Data and Planning of Neurosurgical Procedures. J Digit Imaging. 2021 Aug;34(4):1014-1025. doi: 10.1007/s10278-020-00412-3. Epub 2021 May 23. PMID: 34027587; PMCID: PMC8455790.

[6] Ryo Suzuki, Adnan Karim, Tian Xia, Hooman Hedayati, and Nicolai Marquardt. 2022. Augmented Reality and Robotics: A Survey and Taxonomy for AR-enhanced Human-Robot Interaction and Robotic Interfaces. In Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems (CHI '22). Association for Computing Machinery, New York, NY, USA, Article 553, 1–33. https://doi.org/10.1145/3491102.3517719

[7] Vörös, Viktor, Li, Ruixuan, Davoodi, Ayoob, Wybaillie, Gauthier, Vander Poorten, Emmanuel and Niu, Kenan. (2022). An Augmented Reality-Based Interaction Scheme for Robotic Pedicle Screw Placement. Journal of Imaging. 8. 273. 10.3390/jimaging8100273.

[8] Feig S. Cost-effectiveness of mammography, MRI, and ultrasonography for breast cancer screening. Radiol Clin North Am. 2010 Sep;48(5):879-91. doi: 10.1016/j.rcl.2010.06.002. PMID: 20868891.

[9] Z. Lin et al., "ARei: Augmented-Reality-Assisted Touchless Teleoperated Robot for Endoluminal Intervention," in IEEE/ASME Transactions on Mechatronics, vol. 27, no. 5, pp. 3144-3154, Oct. 2022, doi: 10.1109/TMECH.2021.3105536.

[10] C. M. Morales Mojica et al., "Interactive and Immersive Image-Guided Control of Interventional Manipulators with a Prototype Holographic Interface," 2019 IEEE 19th International Conference on Bioinformatics and Bioengineering (BIBE), Athens, Greece, 2019, pp. 1002-1005, doi: 10.1109/BIBE.2019.00186.

[11] J. D. Velazco-Garcia et al., "Evaluation of how users interface with holographic augmented reality surgical scenes: Interactive planning MR-Guided prostate biopsies," The International Journal of Medical Robotics and Computer Assisted Surgery, vol. 17, no. 5. Wiley, Jun. 08, 2021. doi: 10.1002/rcs.2290.

[12] C. Xue, Y. Qiao and N. Murray, "Enabling Human-Robot-Interaction for Remote Robotic Operation via Augmented Reality," 2020 IEEE 21st International Symposium on "A World of Wireless, Mobile and Multimedia Networks" (WoWMoM), Cork, Ireland, 2020, pp. 194-196, doi: 10.1109/WoWMoM49955.2020.00046.

[13] A. Aristidou and J. Lasenby, "FABRIK: A fast iterative solver for the Inverse Kinematics problem", Graphical Models, vol. 73, no. 5, 2011, pp. 243-260

[14] 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.

[15] M. E. Walker, H. Hedayati and D. Szafir, "Robot Teleoperation with Augmented Reality Virtual Surrogates," 2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI), Daegu, Korea (South), 2019, pp. 202-210, doi: 10.1109/HRI.2019.8673306.

[16] V. Groenhuis and S. Stramigioli, "Rapid Prototyping High-Performance MR Safe Pneumatic Stepper Motors," in IEEE/ASME Transactions on Mechatronics, vol. 23, no. 4, pp. 1843-1853, Aug. 2018, doi: 10.1109/TMECH.2018.2840682.

[17] Raza S, Sekar M, Ong EM, Birdwell RL. Small masses on breast MR: is biopsy necessary? Acad Radiol. 2012 Apr;19(4):412-9. doi: 10.1016/j.acra.2011.12.014. Epub 2012 Jan 24. PMID: 22277636.

[18] El Khouli RH, Macura KJ, Barker PB, Elkady LM, Jacobs MA, Vogel-Claussen J, Bluemke DA. MRI-guided vacuum-assisted breast biopsy: a phantom and patient evaluation of targeting accuracy. J Magn Reson Imaging. 2009 Aug;30(2):424-9. doi: 10.1002/jmri.21831. PMID: 19629977; PMCID: PMC2735014.