Simulating Augmented Reality within Virtual Reality to Prototype User Interfaces That Show the Intent of Robots

Kai Groetenhardt

RheinMain University of Applied Sciences
Faculty of Design, Computer Science, Media
Unter den Eichen 5
65195 Wiesbaden
Germany
kai.groetenhardt@hs-rm.de

Abstract: Since a new generation of augmented reality (AR) head-mounted displays (HMDs) was released, AR is a tool that is increasingly used to communicate robots' intent to users. Still, it is a non-trivial task to implement new AR user interfaces (UIs) especially if it is used to annotate moving objects which makes object tracking a requirement. In comparison, implementing UIs in virtual reality (VR) regarding moving objects is easier since object positions are known within the system. In this article, a prototype of an AR simulation in VR is presented in which two AR-UIs are implemented. Also, the design decisions during the creating process of the prototype are explained. A first experience using the prototype suggested it is possible to gather usable results using an AR simulation in VR. Those results indicated that with one of the AR-UIs it is possible to complete the task, they are designed to accomplish, faster.

Keywords: augmented reality, virtual reality, augmented reality simulation, human-robot interaction, robot behavior, robot intent communication

1 Introduction

An overlapping perceptual space of humans and robots is a key requirement for effective human-robot interaction, which was already identified by Breazeal et al. [BEFS01]. To enhance that overlapping perceptual space, augmented reality (AR) can be used; AR devices enable the robot to show information within the 3D space of the human's view, and the human is able to interact with the user interfaces (UIs) using interaction methods provided by the AR devices. In recent years, a new generation of AR head-mounted displays (HMDs) such as the *Microsoft HoloLens* [MS] and the *Magic Leap 1* [ML] was released, which eliminated some limitations of past AR-HMDs and therefore provide new possibilities for researchers.

However, creating new UIs for AR, especially those involving specific moving objects, can still be difficult and is being researched [HZ19]. For example, to show an UI element besides an object, the object not only needs to be identified first it also has to be tracked, which is

Veröffentlicht durch die Gesellschaft für Informatik e.V. 2020 in B. Weyers, C. Lürig, D. Zielasko (Hrsg.):
GI VR / AR Workshop 2020, 24. - 25, September 2020, Trier.

http://dx.doi.org/10.18420/vrar2020_24

difficult due to problems such as possible full or partial occlusion, a cluttered background, pose variations, or an abrupt motion. The head tracking of the new generation AR-HMDs, on the other hand, works well enough to enable virtual objects to stay at the position they are supposed to be located without much jittering, since the surrounding geometry does not have to be identified first and the geometry usually provides more tracking points. In contrast, virtual reality (VR) systems don't need to track objects at all within the virtual world. The objects' positions are known and can be used without measuring them first. That also applies for simulated robots inside the VR. Therein, virtual robots are not limited by challenges that are given by the need to measure the real world. Also, other things regarding the abilities of robots that are not yet possible in the real world, because of technical limitations, could be simulated in VR to give a glance into the future. Those advantages of VR over AR could be used to quickly prototype AR-UIs in VR. The UIs in VR then can be tested by conducting user studies to determine if it is worth implementing them for an AR system. Simulating AR in VR was researched by Ragan et al. [RWBH09] and Lee et al. [LBHB09]. Its effectiveness was not proven in their articles, but they encourage to further pursue this approach.

To show an example, this article contributes a VR prototype in which two AR-UIs are simulated and a description of the design decisions made while creating the prototype. Both AR-UIs are designed to help users understand a robot's intent regarding objects by showing annotations for those objects, but both AR-UIs use slightly different approaches. In this example, the intent of the robot is to pick up boxes to stack them in a specific order.

This article is organized as follows: In Section 2, the state of the art regarding robots communicating their intent to users with the help of AR is portrayed. Section 3 contains the description of the prototype that simulates AR-UIs in VR. In Section 4, first findings regarding the prototype and its underlying concept are discussed. Section 5 concludes this article and shows possible future work.

2 State of the Art

Communicating a robot's behavior to users via AR is a topic that gains an increased amount of research. This is evident by new specialized workshops taking place covering that topic [WSCBA18], [WSCP19]. In this section, articles are presented that use AR to show the robot's intent to the user. It begins with two articles focused on a robot's movement intent followed by two articles focused on communicating a robot's plan.

2.1 Movement Intent

To show a flying robot's movement intent Walker et al. [WHLS18] developed four different methods using the AR-HMD *Microsoft HoloLens*. These four methods were compared by conducting a user study to see how the displayed virtual imagery affected participants' understanding of robot movement intent, whereby the concept named *NavPoints* ranked best. In *NavPoints* virtual navigation points in the form of spheres get added into the virtual 3D space visible for the user. The points are connected via lines, which indicate the order in which they will be passed by the robot. Above the points two radial timers are getting displayed, which show the robot's arrival and departure time. The concept is visualized in Figure 1 (A).

Setups with flying robots potentially are a safety concern due to the fast spinning rotors that could harm humans. In the article of Walker et al. the topic of safety is briefly mentioned and they ensured to have had safety precautions in place while conducting their tests. In a setup with a higher risk it could be considered to simulate the scene in VR to test UIs that, for example, are meant to ensure the safety of collaborating with a robot, which would eliminate the risk of humans getting harmed by that robot.

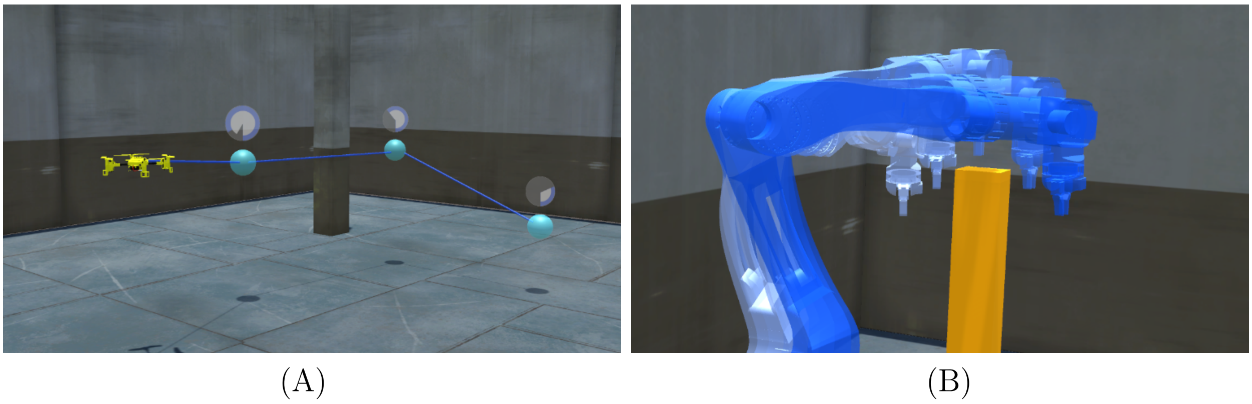


Figure 1: Two different concepts to communicate robot movement intend to users. (A) shows the concept *NavPoints* of Walker et al. [WHLS18] in which connected waypoints, the arrival, and the departure time of a robot can be seen. The image is a rework of Figure 1A in [WHLS18]. In (B) the concept of Rosen et al. [RWP$^+$20] is shown in which several steps of the planned movement of a robot arm are displayed. The image is a rework of Figure 1 in [RWP$^+$20].

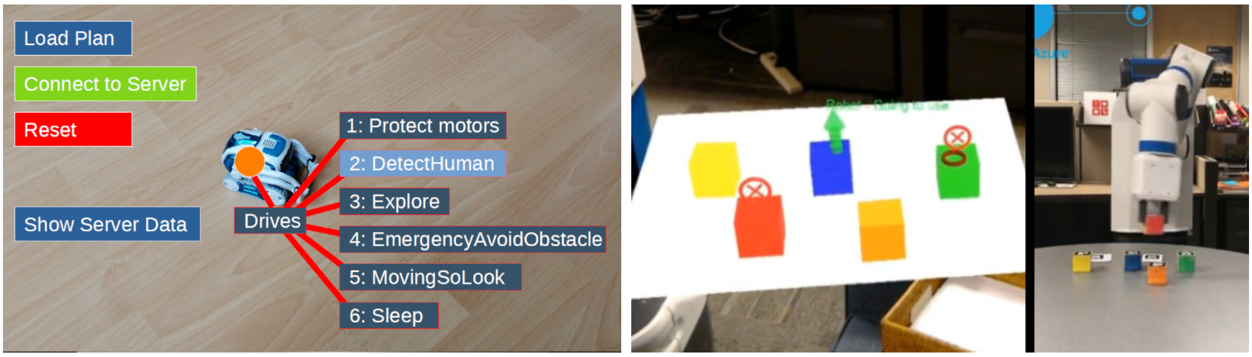
For users to detect collisions of a robot arm with objects surrounding it, Rosen et al. [RWP$^+$20] developed a system using the AR-HMD *Microsoft HoloLens*. Within the 3D space displayed by the AR-HMD several virtual copies of the robot arm get displayed along the path the robot intends to move. The user then is able to see where the virtual robot arms intersect with real objects, which enables the user to predict collisions. This concept can be seen in Figure 1 (B). That system was compared to a 2D version of itself. The 2D version was displayed on a monitor with the possibility to move the virtual camera using mouse and keyboard. Comparing those two, the authors found that the AR version reduced the completion time of the task and increased the average accuracy of collision predictions.

One question not answered in the article of Rosen et al. is why the user has to detect the potential collisions of the robot arm and the objects him or herself and not let the AR system detect the collisions. A possible answer is that the registered geometry by the *Microsoft HoloLens* is not accurate enough to reliably detect collisions. But it would be interesting to see how a UI for this would look like for which a simulation in VR could be considered.

2.2 Plan Communication

To communicate a robot's goals and priorities, Rotsidis et al. [RTBW19] use AR within an *Android* application. As the robot is executing its tasks, a user can point the camera of an handheld device, with the *Android* application running, at the robot, which will trigger a graph to appear. That hierarchical graph is showing the robot's plan with the active task highlighted, which is illustrated in Figure 2 (A). The user is able to expand or fold the graph to see more or less information. The user study, the authors conducted, showed the robot is perceived more alive, livelier, and friendlier with the app than without it.

In the article of Rotsidis et al., it is a rather involved task to see the robots plan, since a device needs to be held and actively pointed at the position the robot is located at. An alternative approach would be to use an AR-HMD instead of a handheld device to enable the user to use his or her hands for other tasks.



(A) (B)

Figure 2: Two different concepts to communicate a robot's intent to a user. In (A), the UI concept of the *Android* AR app of Rotsidis et al. [RTBW19] can be seen that shows the robot annotated with its current plan represented as a hierarchical graph in which the active task "DetectHuman" is highlighted. The image is a rework of Figure 2 in [RTBW19]. (B) shows the concept of Chakraborti et al. [CSKK18]. On the left hand side, the view through the *Microsoft HoloLens* and the virtual copies of the boxes in front of the robot can be seen. The virtual boxes are annotated with symbols that indicate the robot's intent. On the right hand side, the robot is displayed while executing its task of stacking boxes. The image was taken from a video linked in [CSKK18] with kind permission of Tathagata Chakraborti.

Chakraborti et al. [CSKK18] use the AR-HMD *Microsoft HoloLens* to show a robot's intent regarding real objects. In their system, the robot is tasked to stack several boxes. The robot plans ahead which boxes it will use, which is communicated to a human collaborator wearing an AR-HMD. Inside the virtual 3D space, displayed by the AR-HMD, a virtual copy of the boxes is shown and annotated with symbols indicating the robots intent regarding them. A green upwards pointing arrow above a virtual box shows that the corresponding real box will be picket up next. Red circles with a cross inside are marking virtual boxes of which the corresponding real boxes will be picked up later. The user is able to claim boxes for him or herself by selecting the virtual copies of the boxes even if the robot has planned to use them itself. Doing so, the robot changes its plan, which leads to an update of the annotations. The concept of Chakraborti et al. can be seen in Figure 2 (B).

Since the work of Chakraborti et al. is used as a starting point for the prototype presented in this article, it will be discussed in Section 3.

3 A Prototype for Simulating Augmented Reality User Interfaces

The work on user-friendly UIs in AR is partially hindered by the limitations of the currently available technology. Assuming that eventually those limitations will be overcome, it could be beneficial to already have developed UIs that also have been confirmed as user-friendly. To accomplish that, this article proposes to simulate AR in VR to develop and test UIs that currently cannot be created in real AR. This work and the following prototype are to be understood as the foundation for further research in this area.

Each AR concept of the in Section 2 presented articles could theoretically be simulated in a VR system to tweak and test them. As a starting point for this prototype the work of Chakraborti et al. [CSKK18] is used, because the concept of including objects other than just the robot itself in an AR-UI to communicate the robot's intent is for this work the most interesting approach. In the article of Chakraborti et al., the idea is to annotate virtual copies of objects with symbols. Let us hypothesize it would be better, from a user experience (UX) perspective, if the annotations are displayed directly at the real objects. Doing that, the user would not need to switch between two working areas and does not have the additional mental load of connecting the virtual boxes with the real ones. To ease the reference of those two variants, the original concept of Chakraborti et al. in which virtual copies of boxes get annotated will be called *DelegatedAnnotations*, and the modified version of it in which the boxes are annotated directly is named *DirectAnnotations*.

3.1 Scene

The example scene consists of a robot with an arm that enables it to pick up boxes, six differently colored boxes on a table in front of the robot, and a user wearing an AR-HMD. The robot has the task to build a tower of four boxes. When the robot starts its task, it chooses four boxes at random and internally saves them in a list. The first box in the list will be the foundation. The second box in the list is the first box the robot will pick up and then place onto the foundation box. After that, the just placed box becomes the new foundation and the third box in the list will be picked up next. The robot will continue this procedure until the tower is finished. Along the way, the robot communicates the just explained plan to the user via AR by showing following three symbols: 1) A downward pointing arrow annotates a box on which the next box will be stacked on. 2) An upward pointing arrow shows which box will be picked up next. 3) A circle with a cross inside marks boxes that will be used in the future. Also, the user is able to claim a box for him or herself, which leads to the robot deleting that box from its list.

3.2 Augmented Reality Paradigm

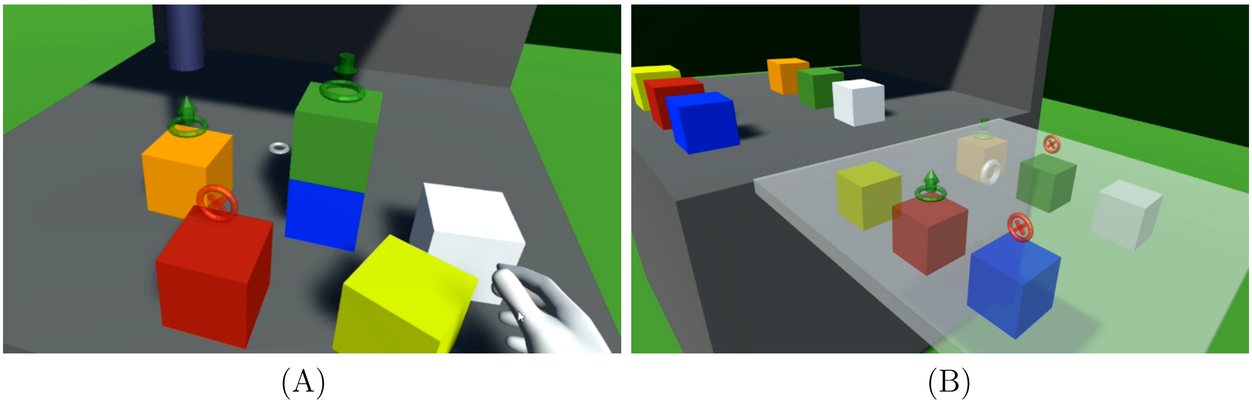
In this VR system, one goal is to mimic the experience of using the AR-HMD *Microsoft HoloLens*. The *Microsoft HoloLens* creates an internal representation of its surrounding geometry to track its own position and to be able to place virtual objects in the virtual 3D space. By knowing the 3D geometry, it is also possible to occlude virtual objects. Using this approach, the user is able to move freely, while the virtual elements seem to be positioned inside the real world.

For the user to be able to point at objects, the *Microsoft HoloLens* uses something similar to a mouse cursor, which is controlled by the user's look direction. The cursor gets displayed in 3D space and is positioned at the surrounding geometry or virtual elements the user is looking at. To trigger actions, the user has to position the 3D cursor at intractable elements and perform a gesture called *Air Tap*. *Air Tap* is executed by pinching the index finger and the thumb together.

How the just explained properties of the *Microsoft HoloLens* are mimicked will be explained in the following sections.

3.3 Setup and Virtual Environment

The VR setup consists of the VR-HMD *Oculus Rift S*, including the *Oculus Touch Controllers*, driven by a high-performance PC that consistently outputs 90 frames per second. The virtual environment was created using the game engine *Unity*. To keep the time investment minimal, a realistic replica of a robot arm or a realistic environment was not created. Instead, the robot arm is portrayed as a simple cylinder with the ability to move and pick up boxes, and the environment is built from simple and plain building blocks. To simulate the 3D cursor of the *Microsoft HoloLens*, the VR system mimics that cursor by positioning a little torus at the position the user is looking at.



(A) (B)

Figure 3: Two scenes are shown in which AR is simulated within VR. In (A) a robot shows its intent by annotating objects with symbols directly. In (B), instead of annotating object directly, a virtual copy of them gets annotated to show the robot's intent, similar to the concept of Chakraborti et al. [CSKK18].

It is important for the user to see which visual elements belong to the UI and to not mistake them for elements that are part of the environment. In VR, all elements are virtual and without visual cues UI elements and objects of the environment look equally virtual. This could have benefits in other scenarios, which will be discussed in Section 4.3, but in this case it is a problem for which a solution was needed. Therefore, all elements of both simulated AR-UIs are displayed with noticeable translucency. This means that in the *DirectAnnotations*-UI the annotations and the 3D cursor are translucent and in the *DelegatedAnnotations*-UI the virtual copies of the boxes, the annotations, and the 3D cursor are translucent. Both variants can be seen in Figure 3.

3.4 Movement and Interaction

To move and rotate the virtual camera within the virtual world, the tracked movements and rotations of the VR-HMD in the real world get transferred to the VR system, which means for the user to move in VR he or she has to move in the real world. Also, the user has the ability to move two virtual hands by moving the *Oculus Touch Controllers*. By touching boxes with the virtual hands in the virtual world while pressing the hand trigger of the *Oculus Touch Controllers* it is possible to grab boxes and attach them to the virtual hand. By releasing the trigger, the attached box gets released from the virtual hand. This enables the users to stack boxes themselves.

Instead of the *Air Tap* gesture used with the *Microsoft HoloLens*, in this prototype the index trigger of the *Oculus Touch Controllers* combined with the look direction is used to trigger an action. This interaction is called *Look&Tap*. If the robot has marked a box to pick it up later, the user still has a chance to claim that box for him or herself. Doing that, the robot will select another box instead to pick it up later. In the *DirectAnnotations*-UI, the user can use the *Look&Tap* interaction with a box to claim it and is then able to pick it up without the robot trying the same. In the *DelegatedAnnotations*-UI *Look&Tap* can also be used, but instead of the box the virtual copy of the box has to be interacted with to claim the corresponding box.

3.5 First experience

To gain first insights, both implemented UI variants were tried by one user. The user is male, 32 years old, uses the *Oculus Rift S* regularly, used the *Microsoft HoloLens* several times and had some experience with toy robots such as the *Anki COZMO* or the *UBTech Jimu Astrobot*. The user had the assignment to claim a box the robot wants to use later and stack it onto an unclaimed box. To do that, the user had do complete the following steps:

1. Identify a box the robot plans to pick up
2. Claim that box
3. Pick up that box
4. Identify an unclaimed box

5. Place the picked up box onto the unclaimed box

With each UI, the user had to complete the assignment 10 times. The VR system measured the time between step 2 and 3, which resulted in an average time of 1.9 seconds for the *DelegatedAnnotations*-UI and 0.6 seconds for the *DirectAnnotations*-UI. The VR system also measured how long it took the user to get from step 3 to 5, which in average resulted in 1.6 seconds for the *DelegatedAnnotations*-UI and 0.6 seconds for the *DirectAnnotations*-UI. Also to mention is that the fastest completion times with the *DelegatedAnnotations*-UI were still slower than the slowest completion times with the *DirectAnnotations*-UI.

4 Discussion

This section starts with the discussion on the gathered results, followed by criticism of the created prototype, and is finished by an exploration of advantages and disadvantages of simulating AR in VR.

4.1 Results

The results gathered from the first experience were not surprising. The longer time period between claiming a box and picking it up with the *DelegatedAnnotations*-UI is probably explained due to the necessity to first examine the virtual copies of boxes, searching the right symbol, and claiming the box before then looking back to the work area with the physical boxes and identify the just claimed box to pick it up. The reason for the longer time period between picking up a box and placing it onto another box with the *DelegatedAnnotations*-UI seems to be the need to look back to the virtual copies of the boxes to identify a box that is free before the currently held box can be placed.

Those results also matched how the user perceived the duration of completing his assignment. He commented that he felt to have completed the assignment "much faster" with the *DirectAnnotations*-UI because he did not have to switch work areas. The *DelegatedAnnotations*-UI he described as "overly complicated", because he had to first find the virtual representation of the box he wanted to claim, and after that switch the work area and find the real box as well to pick it up.

4.2 Prototype

To further mimic the use of a *Microsoft HoloLens* in the future, it would be possible to use a hand-tracking device, which enables the users to use the *Air Tap* gesture. A hand tracking device could also eliminate the need of using the *Oculus Touch Controllers* to grab objects. Also, the reduced field of view of a *Microsoft HoloLens* could be simulated.

Currently, when the user claims a box for him or herself the respective symbol just disappears. To make the change of state more prominent, some kind of animation for the symbol could be added. Also, instead of just showing which boxes the robot will pick up in the future the order could be visualized. Doing that, the user would be able to create a mental picture of how the box tower will look like before it is finished.

The effectiveness of the chosen method to visualize virtual UI elements by making the translucent needs to be verified. To do that, a user study could be conducted in which users have to identify all elements that are supposed to be virtual. Also, it should be tested if the used symbols can be understood without an explanation. The user knew the meaning of the symbols before he first tried the UIs.

4.3 Simulating Augmented Reality

In this article, some of the limitations of the current AR technology were mimicked, but not having those limitation in VR could also be utilized. In AR, it is currently not possible to perfectly integrate a virtual object in the real world in a way that a user does not recognize that object as virtual. This is caused by limitations in the hardware and by unsolved issues in the rendering process of virtual objects. For an object to appear real, it needs to receive and cast shadows, its reflection should match its surroundings of the real world, light should bounce off of it and change the color of surrounding objects, and more. In VR, all those issues do not exist, since the environment and the simulated virtual objects are part of the same virtual world and are being rendered in the same way. Therefore, the visual style of augmented virtual objects and the rest of the virtual world match perfectly. This circumstance could be used to test concepts of AR systems in which the perfect integration of an augmented virtual object is a requirement.

Simulating AR in VR does not only have advantages. There are some challenges involved, which have to be considered before building a simulation. For a simulation, parts of the real world have to be modeled which is not limited to the visuals and function, but can also include sound, haptic, smell, and more. Depending on the needed detail this can be very time-consuming, also the maximum detail of a virtual world is limited by the current technology. For the feel of presence in a virtual world the body of the user, or at least parts of the body, need to be simulated. This is not limited to the movement of the body but should also include a representation of the body which can be done in different variations (e.g. only virtual hands, only virtual hands and virtual arms, etc.). Depending on how the virtual body is visualized, there can be a variance in the feel of presence for different users. For example, Schwind et al. [SKT$^+$17] showed that the feel of presence for men is reduced by non-human hands compared to models of male or female human hands. For women, the feel of presence is reduces by having virtual male human hands compared to non-human hands.

However, the goal of this kind of AR simulation is not to have the most realistic and most immersive VR system that enables the most intense feel of presence. The simulation only needs to meet the minimum requirements to be able to test different UIs. Likely, these minimum requirements differ from use-case to use-case and should be determined by early involving users to test the system. The first experience with the presented prototype indicates that it is possible to extract data from an AR simulation in an easy to setup VR system that can be used to measure completion times of interactions and to identify flaws in UI designs.

5 Conclusion and Future Work

In this article, two AR-UIs were simulated and compared within a VR system. It was possible to gather time measurements while having a first experience with the system and identify a potential flaw in one of the UI concepts. The results indicated it is advantageous to annotate real objects directly instead of annotating virtual copies of them, since switching work areas to look at the copies takes extra time.

The immediate next step is to conduct a representative user study comparing the two UI concepts to confirm the findings of the first experience with the prototype. Since humans collaborating with robots to stack boxes is not a realistic scenario, a realistic use case should be selected for which elements of the presented concept could be adapted and to further research the concept of simulating AR in VR.

References

[BEFS01] Cynthia Breazeal, Aaron Edsinger, Paul Fitzpatrick, and Brian Scassellati. Active vision for sociable robots. *IEEE Transactions on systems, man, and cybernetics-part A: Systems and Humans*, 31(5):443–453, 2001.

[CSKK18] Tathagata Chakraborti, Sarath Sreedharan, Anagha Kulkarni, and Subbarao Kambhampati. Projection-aware task planning and execution for human-in-the-loop operation of robots in a mixed-reality workspace. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4476–4482. IEEE, 2018.

[HZ19] Pengfei Han and Gang Zhao. A review of edge-based 3d tracking of rigid objects. *Virtual Reality & Intelligent Hardware*, 1(6):580–596, 2019.

[LBHB09] Cha Lee, Scott Bonebrake, Tobias Hollerer, and Doug A Bowman. A replication study testing the validity of ar simulation in vr for controlled experiments. In *2009 8th IEEE International Symposium on Mixed and Augmented Reality*, pages 203–204. IEEE, 2009.

[ML] Spatial computing for enterprise | magic leap. `https://www.magicleap.com/en-us`. accessed: 07/04/2020.

[MS] Microsoft hololens | mixed reality technology for business. `https://www.microsoft.com/en-us/hololens`. accessed: 07/04/2020.

[RTBW19] Alexandros Rotsidis, Andreas Theodorou, Joanna J Bryson, and Robert H Wortham. Improving robot transparency: An investigation with mobile augmented reality. In *2019 28th IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*, pages 1–8. IEEE, 2019.

[RWBH09] Eric Ragan, Curtis Wilkes, Doug Bowman, and Tobias Höllerer. Simulation of augmented reality systems in purely virtual environments. *Virtual Reality Conference, IEEE*, 0:287–288, 03 2009.

[RWP+20] Eric Rosen, David Whitney, Elizabeth Phillips, Gary Chien, James Tompkin, George Konidaris, and Stefanie Tellex. Communicating robot arm motion intent through mixed reality head-mounted displays. In *Robotics Research*, pages 301–316. Springer, 2020.

[SKT+17] Valentin Schwind, Pascal Knierim, Cagri Tasci, Patrick Franczak, Nico Haas, and Niels Henze. "these are not my hands!" effect of gender on the perception of avatar hands in virtual reality. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, pages 1577–1582, 2017.

[WHLS18] Michael Walker, Hooman Hedayati, Jennifer Lee, and Daniel Szafir. Communicating robot motion intent with augmented reality. In *Proceedings of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '18, pages 316–324, New York, NY, USA, 2018. Association for Computing Machinery.

[WSCBA18] Tom Williams, Daniel Szafir, Tathagata Chakraborti, and Heni Ben Amor. Virtual, augmented, and mixed reality for human-robot interaction. In *Companion of the 2018 ACM/IEEE International Conference on Human-Robot Interaction*, HRI '18, pages 403–404, New York, NY, USA, 2018. Association for Computing Machinery.

[WSCP19] Tom Williams, Daniel Szafir, Tathagata Chakraborti, and Elizabeth Phillips. Virtual, augmented, and mixed reality for human-robot interaction (vam-hri). In *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 671–672, 2019.