AY 2018

WeARable:
An Investigation into the Integration of Multipurpose Wearable Robotics and Augmented Reality

A Thesis Submitted to the
Department of Computer Science and Communications Engineering,
the Graduate School of Fundamental Science and Engineering of
Waseda University
in Partial Fulfillment of the Requirements for the
Degree of Master of Engineering

July 24th, 2018

By:
Jaryd Ashley Urbani
(5116FG10-2)

ADVISOR: PROF. TATSUO NAKAJIMA

Table of Contents

ABSTRACT

Wearable robotics is an emerging class of devices that utilize shape-changing capabilities to facilitate functionality in a variety of interaction contexts. The operational possibilities made available by such devices are highly intriguing, though research into the interoperability of wearable robots with other forms of wearable devices remains largely uninvestigated. We introduce "WeARable" a preliminary integration framework in which Augmented Reality (AR) is utilized to compliment the interaction capabilities of such wearable robots. We modeled the framework around a chosen design space in order to address four main aspects of interaction: Control and Feedback, SRL Operations and Capabilities, Always Available Input, and a Digital Agent.

The framework consists of our AR user interface, robot controller, and AR-robot integration software, and we demonstrate our framework utilizing an example prototype wearable robot. Integration is accomplished by publisher-subscriber model, in which robot functionalities are exposed as services on a network. Other sub-systems are then able to access these services and execute required robot functionality or retrieve device information.

We also demonstrate the feasibility of our framework with a comprehensive user experience solution. The solution includes a number of example applications, which aim to address each of the domains within our design space. This solution shows how AR could be used in controlling, providing feedback, and visualizing states and statuses of the wearable devices and their functionality.

We evaluate our implemented solution by means of a user study involving a number of student participants. Results indicated that these participants found our approach innovative, but highlighted some challenges experienced with AR tracking when interacting within some of the embedded applications. Lastly, we discuss the limitations we encountered during our approach and provide possible future research direction.

ACKNOWLEDGEMENT

I would first like to thank my thesis advisor Prof. Tatsuo Nakajima of the Department of Computer Science and Engineering at Waseda University. The door to Prof. Nakajima's office was always open whenever I ran into a trouble spot or had a question about my research or writing. He consistently allowed this paper to be my own work, but steered me in the right direction whenever he thought I needed it.

I would also like to acknowledge Mr. Mohammed Al-Sada of the Department of Computer Science and Engineering at Waseda University as the second reader of this thesis, and I am gratefully indebted to him for his very valuable comments on this thesis.

The presented work was supported in part through the "Master's Degree and Internship Program of African Business Education Initiative for Youth" (ABE Initiative) provided by the Japan International Cooperation Agency (JICA), the Distributed and Ubiquitous Computing Lab (DCL) of Waseda University, and the Program for Leading Graduate Schools, "Graduate Program for Embodiment Informatics" by the Ministry of Education, Culture, Sports, Science and Technology. We acknowledge the financial support from KAUTE Foundation for the research reported in this paper.

Finally, I must express my very profound gratitude to my parents for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

Author

Jaryd Ashley Urbani

CHAPTER 1. INTRODUCTION

Wearable robotics is an emerging class of devices, which has since been researched in various communities. Overall, robotic actuation provides a variety of interesting interaction potentials, ranging from haptic controls and shape-changing interfaces to Supernumerary Robotic Limbs (SRL's) or exoskeletons that can empower user's limbs. However, despite most wearables' innovative designs, interaction with these devices is an ongoing challenge, most lack the input and feedback control methods to compliment actuation and shape-changing operations, or are comprised solely of interaction methods for a singular usage context [1], [2], [3]. Further challenges we confront include the ability for wearable actuated robotic devices to be used independently of the users' physical limbs and/or appendages, and for the entire system to be operational whilst on the move.

In order to address these challenges, we developed a framework in which we utilize Augmented Reality (AR) interaction technologies and demonstrate how AR can bridge the gap between user and the actuated wearable device via a Head Mounted Display (HMD). The integration of AR and wearable robotics presents a highly intriguing potential interaction medium, which remains largely unexplored, especially cross-device interaction that involves multiple wearable devices within a mobile context. Recent advancements in the field of AR have made the technology easily accessible and a popular platform for a multitude of applications. Modern AR platforms, such as smartphones or HMD's, also include several sensors and input modalities that enrich the entire AR experience.

Accordingly, we introduce "WeARable", a preliminary framework for the integration of AR and wearable robots, and the development of functional and coherent AR-robot applications and user experiences. The framework consists of our AR user interface, robot controller, and AR-robot integration solution, and we demonstrate our framework utilizing an example prototype wearable robot.

Our wearable prototype is a wrist-worn snake type device. We selected this robot configuration to allow for the maximum number of possible use cases, it also allows for a number of basic functionalities to be executed whilst the device detached from the user.

Figure 1 : The WeARable project.

The WeARable framework offers the flexibility of both AR and robot control systems. These are implemented via a number of publisher-subscriber model services made available over the network. Other systems or applications are then able to connect to these services and execute robot commands or retrieve device information.

Our framework was designed and developed to cater to four main aspects of AR and wearable device interactions. These are defined within our design space as the 1) Control and Feedback domain, 2) SRL Operations and Capabilities domain, 3) Always Available Input domain, and 4) a Digital Agent or 'Assistant' domain.

To demonstrate our framework feasibility, we developed a comprehensive user experience solution. This solution includes a number of example embedded applications aimed to address each of these design space domains and demonstrate some possible system functionalities.

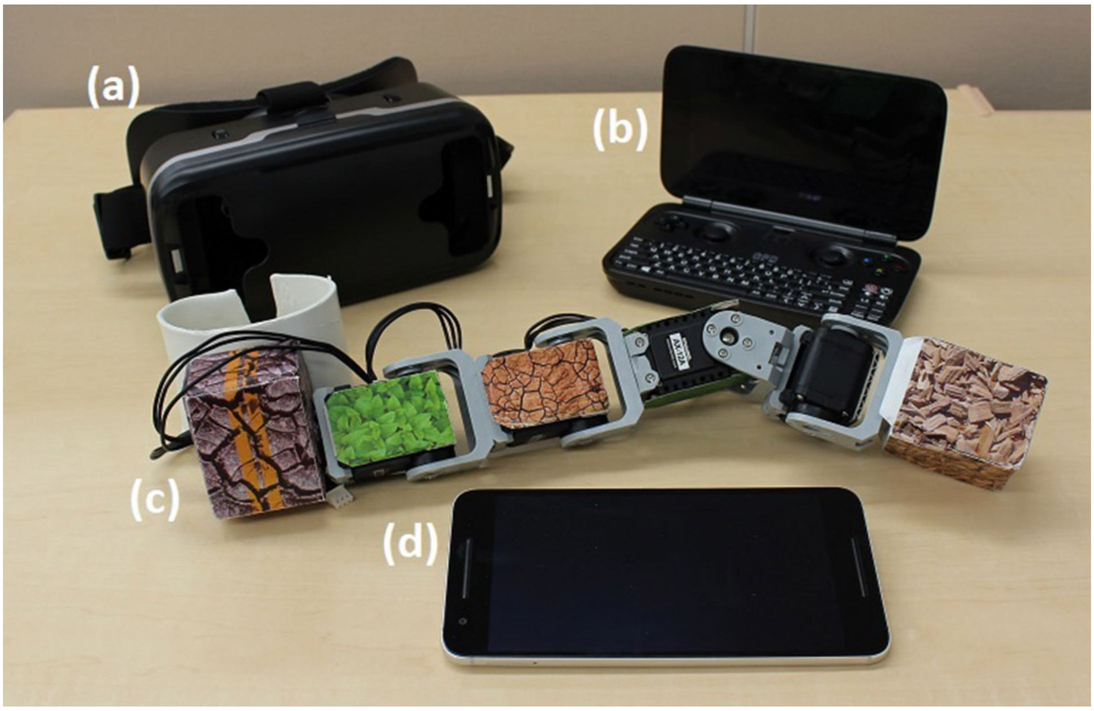

Figure 2 : Fundamental components for our solution: (a) AR enclosure, (b) GPD WIN robot controller, (c) wearable robot with embedded AR markers, and (d) Android smartphone.

We conducted a user study to evaluate our implemented solution. A number of students participated and shared their opinions on the usability of the system, also quantifying their experience as a whole. Results indicate that the majority of participants favoured the agent application, for both novelty and usefulness, whilst the device status function was favoured least.

Through this research, we are able to contribute a flexible framework that enables developers the ability to build applications and user experiences, involving wearable robots and AR, effectively and efficiently. We provide number of experiences to demonstrate the capabilities of our framework, and present the results of the user evaluations of these experiences. We also discuss the current limitations and possible future direction for our research.

CHAPTER 2. RELATED WORK

Section 2.1 AUGMENTED REALITY

AR allows for digital information, such as text, graphics, or 3-D models, to be rendered within real world contexts. As AR is essentially an output method, AR is supplemented with various interaction modalities [4], [5], such as voice commands or hand gestures. Researchers have investigated using AR for maintenance [6], [7], where visual and auditory instructions can be easily relayed to users in the field. Reality Editor [8] and Smarter Objects [9], show how AR can serve as an effective medium for controlling smart environments. For example, visualization of mappings among smart objects and making alterations to suit their control requirements. Their work also showed how users could control smart objects with AR using a variety of interaction methods.

Section 2.2 AUGMENTED REALITY IN ROBOTICS

Previous work in this subject investigated the use of AR to compliment interaction with generic robots. Several works utilized AR for displaying *robot intention*, whereby the expected motion trajectory or future state of a robot is represented digitally before execution [10], [11], [12], and for *robot control*, where robots can be operated or tele-operated in real-time [8], [13], [14], and for *environmental tracking*, in which AR monitors robots and their environments to facilitate movement and interaction with their surroundings [15], [16], [17]. Some research into digital augmentation of robot form provided robots with previously unavailable functionality such as robot gestures or deixis [18], [19], [20]. With this, anthropomorphic aesthetics, such as faces or hands, can be added to robots to allow for more human-like interaction.

We differ from previous research by focusing on the previously uninvestigated integration of AR with personal use wearable robots. Another challenge our framework addresses is the integration of various types of interaction modalities of AR with multipurpose shape-shifting robots.

.

Section 2.3 Actuated Wearables

Wearables utilize actuation for a variety of purposes. Leigh and Maes [1] demonstrated a wrist-worn shape-shifting robot capable of serving as a supernumerary robotic limb to hold objects, or as a haptic input device for interacting with a PC. This robot is controlled by a 'Myo' electromyography armband [21] that monitors hand gestures. They also presented a wrist-worn robot with interchangeable modules [22], the reconfigurations of which allow for different experiences. These experiences include examples such as notifications, haptic or shape-changing capabilities, or PC controls through a knob type input module. Rovables [2] are swarm type wearable robots that offer a variety of interactions. These robots can reposition themselves, which allows them to deliver haptic feedback in different locations on the user's body. When equipped with LEDs, they can deliver visual notifications or serve as fashion displays [3].

Within surveyed works, cross-contextual control and feedback remains a challenge. Demonstrated scenarios function on a singular interaction method and lack the ability to easily switch between different applications. For example, switching from functioning as a haptic feedback device to a device for self-expression. In contrast to previous works, our framework enables investigation into cross-contextual AR based interaction with multipurpose wearable robots, which we believe was largely unexplored.

Section 2.4 Supernumerary Robotic Limbs

SRL's are a sub-category of wearable robots that relates to human augmentation. Robot technologies are able to enhance a human body with extra 'limbs' with which to a user may interact with their environment, such as arms, legs, hands, or fingers. A number of device control methodologies are investigated within this field, such as foot [23], [24] or head [25] controls based on the movements of the user's appendages, or electromyography [1], [26] controls based on muscle contraction signals.

In comparison, with our integration of AR technology and multipurpose wearable robots, we are able to interact with the device without otherwise occupying the user's limbs or body.

Section 2.5 SHAPE-CHANGING DEVICES

There are multiple interaction capabilities made available by shape-changing devices. Some devices have been utilized for haptic and tactile interfaces [27], [28], or shape-changing notifications [29], others provide dynamic tangible interactions for mobile devices [30], [31], and some are even able to convey or extend natural body-language [32], [33] or emotional [34] cues during user conversation. LineForm [35] and ChainForm [36] demonstrate shape-changing interfaces that can provide a wide variety of use cases, such as tangible user interfaces, haptic feedback, limb empowerment, etc. Thanks to its modular design, ChainForm is also able to adapt to fit various additional operations. These devices are both constructed in a snake-like form factor and are somewhat wearable by the user.

CHAPTER 3. APPROACH AND IMPLEMENTATION

Section 3.1 DESIGN AND PLANNING

Our objective is to provide developers with a framework that allows for the efficient development of integrated AR and wearable robot applications and experiences.

Our design space defines the aspects of interaction we planned to address, which in turn outlines some possible capabilities made available by the framework. Example applications and user experiences were developed to demonstrate these interaction aspects and evaluate the feasibility of the framework (Figure 3).

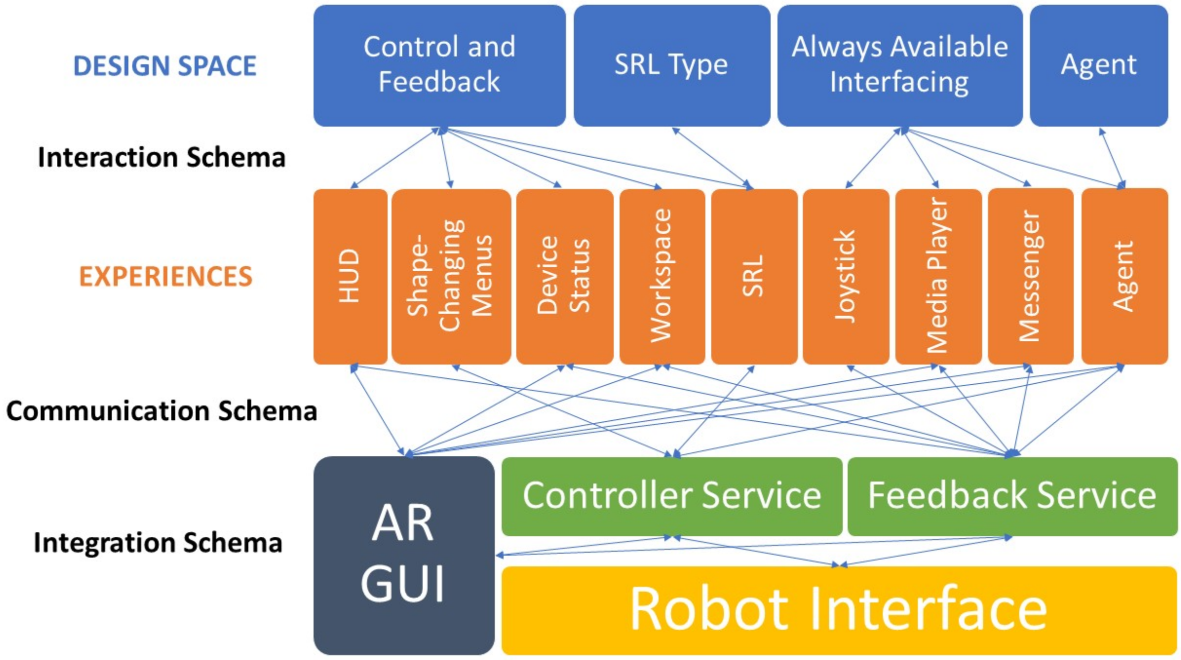

Figure 3 : Overview of interaction, communication and integration schemas within our solution, depicting the correlation between design space, example applications and the framework.

The approach for each component within our solution, including hardware, software and integration implementation is detailed in the sections to follow below.

Section 3.2 DESIGN SPACE

The design space defines the aspects of interaction we planned to address within our solution, and thus outlines some of the possible capabilities made available by our framework. A key factor of our implementation is that the wearable device should be multipurpose, capable of performing numerous day-to-day interactions and operations, both with the user and the surrounding environment.

The use of AR, and related interaction modalities, enables the robot to function independently from the user's physical limbs, which allows the user better multitasking abilities and maximizes practicality and convenience. Without the need for complex control modalities or large control rigs, our integration allows for the entire system to be used in a mobile context. Our framework also enables control and feedback methods from both the wearable robot and AR HMD.

Figure 3 shows to which design spaces each application correlates to, as well as to which framework services they connect for integration and operation.

In order to achieve the desired outcomes of our framework, our design space would have to accommodate numerous AR-robot interaction capabilities. We thus gathered a number of interesting interaction concepts, extracted from numerous use cases found within previous research showing positive potential, together with our own collection of novel interaction concepts. We then adapted and partitioned this collection of interaction concepts to form our intended design space.

The four resulting aspects of integration are defined as follows:

1. *Control and Feedback* of the system via AR and haptic interfaces.
2. *SRL* type capabilities, where the device is used to interact with the user's environment.
3. *Always Available Interfacing*, where the device is used as a controller for external systems.
4. An *Agent*, where a digital avatar is given physical form to perform 'assistant' operations.

Each domain is further detailed, including defined examples, to follow in Chapter 4.

Section 3.3 SYSTEM OVERVIEW

Our framework essentially provides two concepts of operation; 1) robot control via AR interaction, and 2) AR control via robot interaction. We illustrate these concepts in Figure 4 below. The depicted workflows run in numeric order from (1) to (6), showing operational concept 1, or in reverse numeric order from (6) to (3), showing operational concept 2.

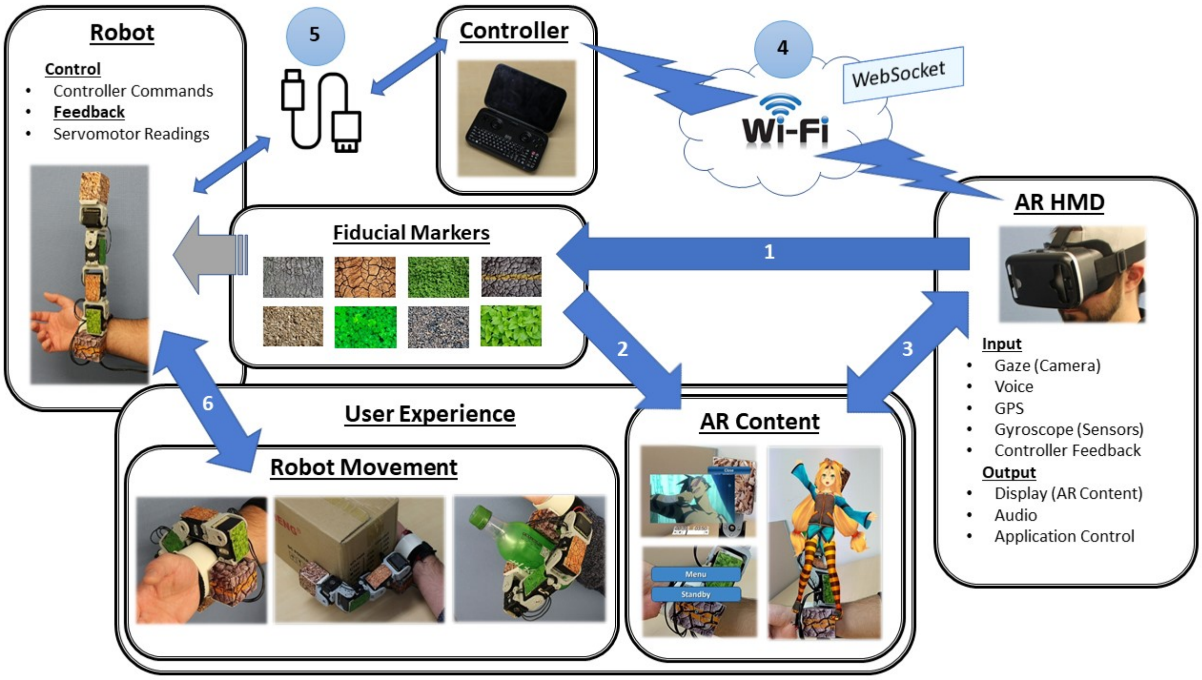

Figure 4 : System operational workflow overview.

An example of the concept 1 workflow would be:

(1) The AR HMD recognizes a fiducial marker.

(2) AR content is displayed. Content is spatially registered to the marker.

(3) AR content command or instruction is selected.

(4) The command is transmitted to the robot control unit over Wi-Fi.

(5) The controller communicates the respective operation to the robot.

(6) The robot executes the operation (Executes motion or assumes pose).

Section 3.4 AR AND HMD

Hardware: The HMD consists of a Google Nexus 6P Android Smartphone fitted within a generic 3-D AR/VR Headset enclosure (Figure 5). The foremost cover of the HMD was removed to allow the smartphone camera to be used for our 'video see-through' type AR application.

Figure 5 : Smartphone and enclosure used for AR HMD.

Software: The AR application was developed using Unity3D [37] and Vuforia 7 [38] and is deployed via Android SDK to our smartphone running Android 8 (Oreo) operating system. The AR user interface is operated by gaze, allowing the user to simply look at objects in order to interact with them. The gaze technique is configured to execute operations once the user has focused on such a control object for two seconds. We utilize fiducial markers for the AR spatial registration (Figure 6), these are recognized by the AR device and provide a fixed point of reference, including size and scale, upon which content may be placed. Fiducial markers are widely used due to their robustness and ease of use. Another advantage of using an HMD is the Head-Up Display (HUD), although technically not an AR technology, HUD refers to digital content which is fixed within the user's field of view and not to the user's environment, as shown in Figure 10. The use of both modalities allows our system to accommodate a variety of interaction experiences.

Section 3.5 WEARABLE ROBOT

Hardware: The robot used is a snake-shaped prototype wrist-worn robot and consists of six interlinked Robotis Dynamixel AX-12A servomotors (stall torque 1.5 Nm) [39] fastened together through AX-12A PLA brackets (Figure 6). This robot configuration was selected to maximize the number of useful robot poses, allowing for the robot to be used in a variety of different applications.

Figure 6 : Prototype robot design.

The total weight of the robot is 500g and the length at full reach is 300 mm. The robot is powered by a lithium polymer battery, allowing for approximately 25 minutes of operation. We fit the robot to the user's wrist via a moulded plastic bracelet, fastened by a Velcro strap providing a stable base. When not in use, the robot wraps around the user's wrist for a more unobtrusive experience (Figure 7).

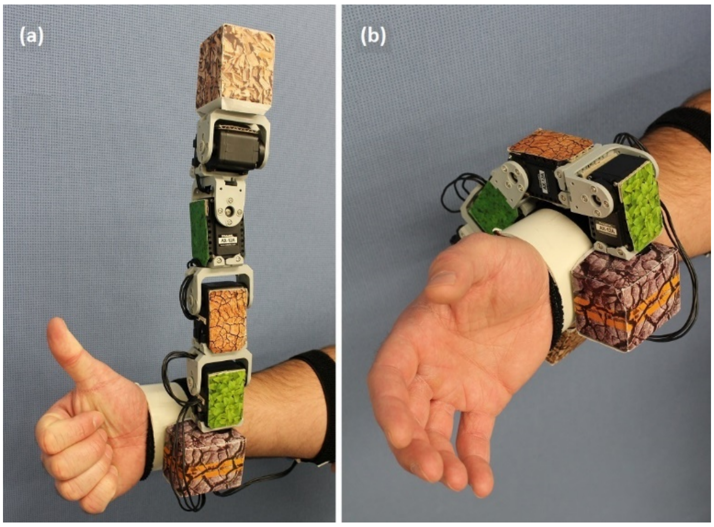

Figure 7 : Robot, as worn by user, is displayed as it appears when it is (a) in use, and (b) not in use.

Software: Our robot control software was developed in using C# utilizing the Robotis SDK [39]. The software handles all robot control and feedback functionality, and provides the interface for the communication of this information between the controller and the AR interface software.

We developed a graphical user interface (GUI) that enables the ability to monitor and control the robot, or any individual servomotor, directly from the controller itself (Figure 8). This allows for easy diagnosis and detailed troubleshooting of the robot or individual servomotor should the need arise.

All available attributes and functionality are provided as services on the network. The system is able to connect and operate on any provided Wi-Fi network.

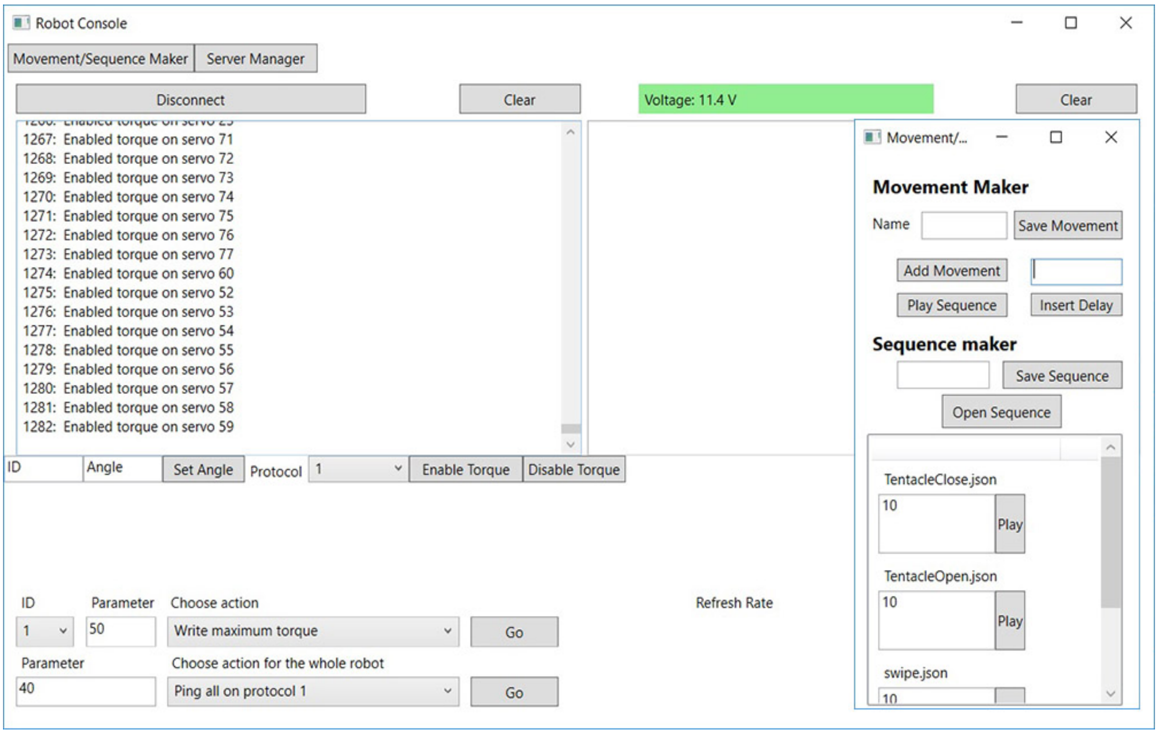

Figure 8 : The robot control software GUI.

Control Unit: The control unit hosts the robot control software and connects to the robot via a USB/3-pin servo controller interface cable. The control unit itself is a GPD WIN portable mini-computer [40] (Figure 2(b)), and runs a version of Windows 10.

Section 3.6 FRAMEWORK INTEGRATION

The framework accomplishes integration between the AR and robot systems by means of a publisher-subscriber service model, which is implemented utilizing the WebSocket communication protocol [41]. The robot control unit acts as the publisher and hosts the services that expose the embedded robot functionality. The AR system connects to these services to access and invoke the defined functionalities and thus fulfils the role of subscriber. For this framework example, we enable two WebSocket services to allow the AR system to issue commands for the robot and receive information about the robot. We categorize the two services as follows:

1) The control service, to which clients may connect in order to invoke servomotor controls, such as rotating servos to specific angles, setting rotation speed, or adjusting torque limits. Changes to robot shape can also be executed in singular or sequential fashion.

2) The feedback service, to which clients may connect in order to receive robot information broadcast at specific rates or on events. Information such as servomotor angles, temperatures, torque limits, and loads are available to all connected clients.

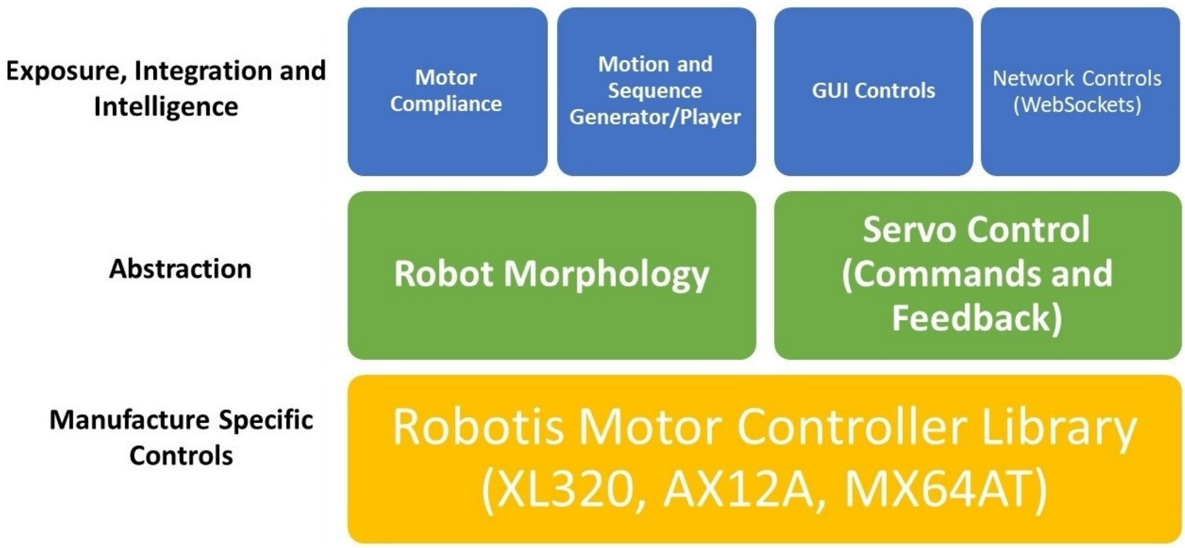

Figure 9 : Integration and robot control hierarchy.

Service messages are formatted in JavaScript object notation (JSON) [42]. Information such as message type, intended command, servomotor IDs, and other parameters, are easily and robustly encapsulated for communication. The following is an example control message, this instructs servomotor 4 to rotate to a 90 degree angle at a speed of 50%:

{"Movement": [{ "ID": "4", "protocolVersion": "1", "ServoType": "AX 12A","angle": "90", "speed": "50" }] }

There are numerous advantages provided by the publisher-subscriber model we employed, most notably segregation. With segregation comes scalability, reliability and adaptability. Any number of services are able to be created, each to host any desired control or feedback requirement, and each to run as separate threads or processes [43]. Upgrades or other changes to hardware, such as AR HMD or the robot itself, are easily accommodated and with no adverse effect to the core integration and control software. It is even possible for multiple devices to interface with multiple robots, enabling a flexible platform for experimentation with different robot or entire system configurations.

Section 3.7 Framework Contributions

User control methods of robots is an ongoing challenge [44], especially with regard to wearable robots [1], [22]. Using our framework, we are able to address numerous challenges previously encountered when integrating AR and various robotics technologies. Potential AR use cases have been presented in previous works, but their implementation infrastructure was not detailed [9], [14], [15], [16], [20]. Similar frameworks have also been described, but tend to focus on differing application domains and/or robot variants, such as work within the industrial sector [45]. In contrast, our framework accommodates different interaction and AR tracking methods, and robot morphologies, and targets the creation of applications and experiences for AR with everyday wearable robots. There also exists a robotics middleware, ROS [46], which is mainly concerned with the integration and control of different robotic components, and differs in scope from our framework.

CHAPTER 4. DESIGN SPACE INTERACTION ASPECTS

Section 4.1 CONTROL AND FEEDBACK

The Control and Feedback domain relates to low-level device manipulation and system information display. Information can be made available to the user, at any time, be it via AR display or represented by device shape or pose.

Examples include:

- Low-level device shape manipulation.
- Motion and sequence recorder/replay.
- Device status display in AR.
- A 3-D virtual representation of device.
- Physical representation of system state.
- Device workspace or range set.

Section 4.2 SRL MODE

The SRL Mode domain represents all ways in which the device may interact with the physical environment, similar to and extending the way in which current SRL devices function today.

Examples include:

- Interacting with the environment whilst attached to the body.
- Interacting with the environment whilst not attached to the body.
- Interacting with the environment whilst either in or out of sight.
- The ability to alter the interaction control method.

Section 4.3 ALWAYS AVAILABLE INPUT

The 'Always Available Input' domain relates to the operations or experiences in which the device could be used to interact with digital contents or other software, either embedded within the AR interface or in conjunction with other capable external systems.

Examples include:

- Utilizing the device as a controller for AR applications.
- Utilizing the device as a controller for a 3rd party system.
- Utilizing the device as a controller for embedded applications.
- Haptic feedback from applications.
- Haptic notifications from the system.

Section 4.4 AGENT

The Agent design space represents the mode in which device and AR are combined in such a way as to anthropomorphize the robot device, leading to a form of cyber-physical assistant with which users are able to interact. The Agent itself is capable of physical and digital interactions with users and vice-versa.

Examples include:

- Haptic control of Agent.
- Haptic feedback from Agent.
- Digital control of Agent.
- Digital feedback from Agent.
- System control of Agent.
- System notifications from Agent.

CHAPTER 5. APPLICATIONS AND EXPERIENCES

Section 5.1 HEAD-UP DISPLAY (HUD)

AR HMD's allow for numerous approaches and representations of information in the context of our design space. Not only are we able to display information and interact with digital contents attached to the device whilst we have it in view, but it also allows for us to have desired information and content available to the user even when the physical device is not in view. We refer to the latter scenario as HUD mode.

As shown below (Figure 10), the HUD contains persistent information, such as (a) the interface cursor and (b) situational command instructions, as well as 'on-demand' application specific information, such as (c) a 3-D representation of current robot pose, (d) device status, and (e) digital content application objects, for when the device is moved out of sight. Details of HUD capabilities within individual applications are explained in the explicit application sections below.

Figure 10 : Interface display showing all available Head-Up Display contents enabled.

Section 5.2 DEVICE STATUS

The Device Status application is a system feedback method where device information is relayed to the user via the AR HMD. This application includes two forms of device information relay as shown in Figure 11:

1) The display of detailed information for each servomotor component. Information such as servo rotation, voltage, temperature, and torque values are available for display. This information is also accessible in two further forms; spatially registered or HUD mode. When the device is within view, the user gazes at a desired component and the respective information is overlayed, in AR, above that selected component (a) and (b). When the device is out of view, the details of every available component is fixed to the right of the user's HUD (c).

2) A 3-D model of the current robot hardware configuration (d). This model, displayed in the bottom left of the user's HUD, changes shape along with the physical robot. This enables the user to know the current robot shape or state whenever the device itself is not within view.

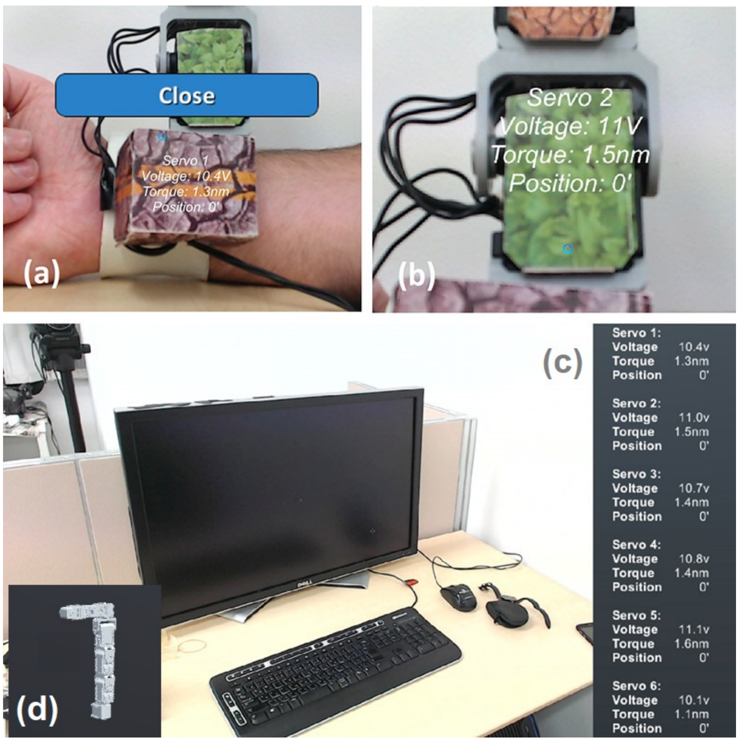

Figure 11 : Device Status information displayed via (a & b) spatial registration, and (c & d) HUD.

Section 5.3 SHAPE-CHANGING MENUS

The Shape-Changing Menus is a system feedback application in which the current system status is reflected physically by changing the device's shape or pose. This provides users with the ability to recognize the state of the device even without explicit AR interfacing.

This application utilizes robot pose control methods to provide a physical representation of the current system state. Example scenarios include; system sleep mode, or menu and application representation. In sleep mode (Figure 7(b)), the device will wrap around the user's wrist to provide minimal obtrusiveness for when the device is not in use. Menu and application representation will change the device shape depending on the current menu iteration or application in use. Figure 12 shows a user iterating through the system menu. Using (e) the cursor, the user browses from (d) the start menu (represented by (a)), through (f) the main menu (represented by (b)), to (g) the app menu (represented by (c)). Similarly, each application has a designated pose representation.

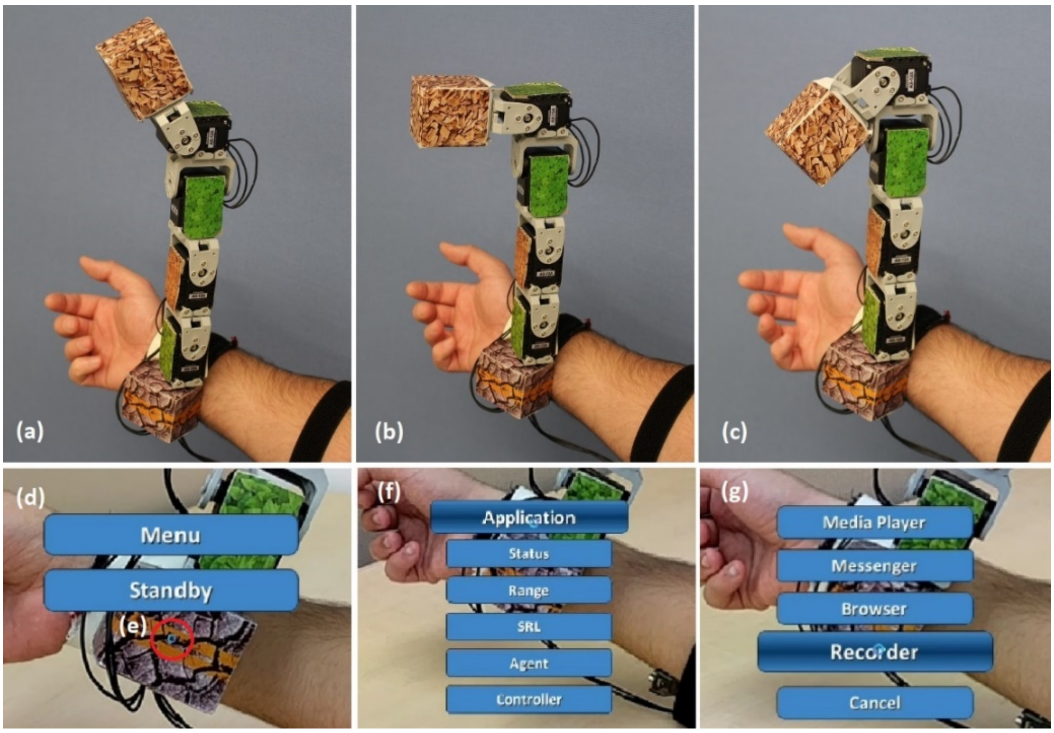

Figure 12 : Example of Menu representations.

Section 5.4 WORKSPACE

The Workspace application is another system feedback method, where the device's range of motion is relayed to the user via the AR HMD. The workspace is displayed as a 3-D semi-transparent sphere, which is scaled and positioned to correspond to a user-selected servomotor.

The application allows the user to select a desired servomotor and view the workspace for that selected servomotor's 'set', as shown in Figure 13. We define the servomotor set as the collection of servomotors starting from the user-selected servomotor to the device's end vector. For example, if the base is selected, the workspace for the entire prototype is displayed (a). Whereas, if the 3rd servomotor from the base is selected, the new workspace is sized and scaled to the 3rd servomotor (b). This is advantageous for scenarios where certain servos would be unable to move or be otherwise inoperable.

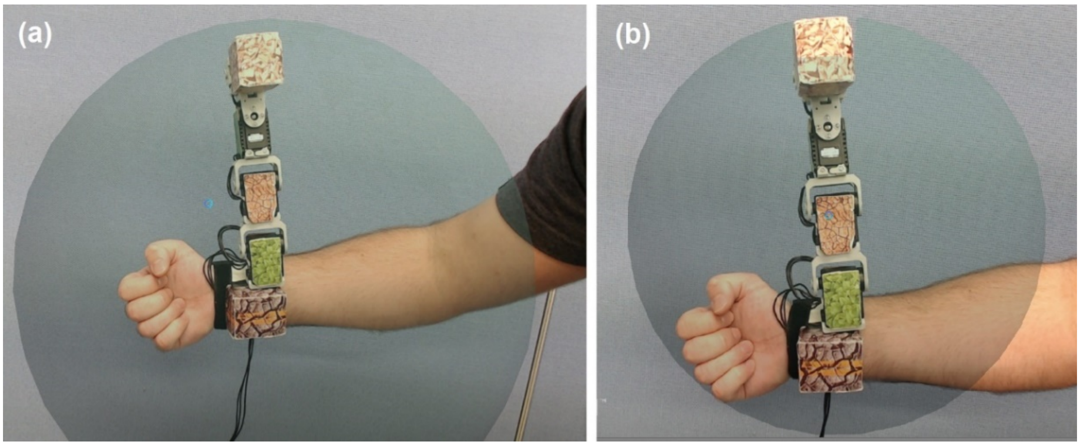

Figure 13 : Workspace display for (a) base, and (b) servo-set 3.

Furthermore, any *recognized* objects which enter into the displayed workspace are highlighted, this clearly demonstrates to the user that the object is within the defined operational range of the robot (Figure 14). Notifications inform the user that the robot is able to interact with said objects (a) and (b), or caution the user that there exists an object which poses a collision hazard (c).

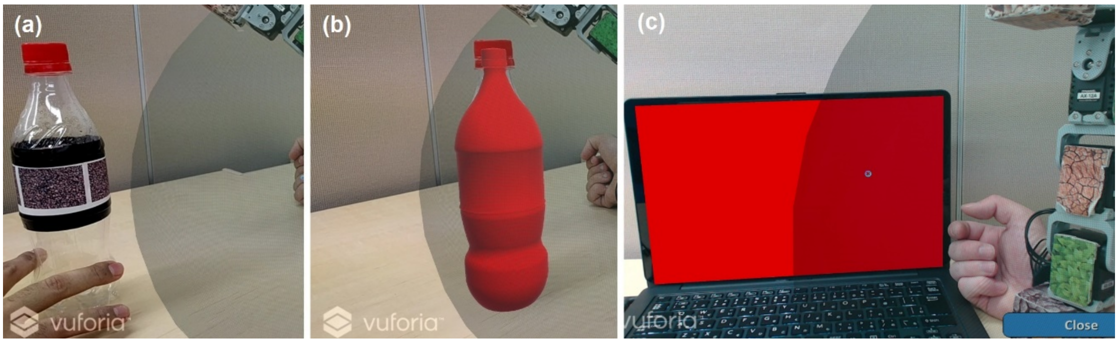

Figure 14 : Example notifications for objects entering workspace.

Section 5.5 DIGITAL CONTENT AND HAPTIC INTERFACING

Digital Content, in this context, refers to a number of embedded applications with which we are able to directly interact. Interaction can occur via the AR interface or by the physical manipulation of the robot device. These include our example Media Player and Messenger applications. By default, the applications are spatially registered to the device's end marker, but the user may also choose to register it instead to any stand alone marker available (i.e. to a desk or wall marker).

Media Player: This application builds a library of any music (*.mp3), image (*.jpg), or video (*.mp4) files stored within the 'media' folder on the Android smartphone. As in Figure 15, the Media Player has two display options; (a) spatial registration with AR interface controls and information panel, or (b) simplified HUD. The HUD mode is automatically triggered when the media player leaves the user's view. Here, the media pins to the user's display border but lacks the control and information panel so as to allow for a less obstructed view of the user's environment.

Messenger: This application allows users to send messages to one another via a hosted 'chat' service on the network. As with the framework integration, communication is also implemented using the WebSockets protocol. There is no limit to the number of users able to join each session. Message composition, in our system, requires a Bluetooth keyboard, until such time as voice or gesture typing becomes a feasible option. Messages are formatted into a "Sent" and "Received" column design (Figure 15(c)).

Figure 15 : Digital Content example applications (a & b) Media Player, and (c) Messenger.

Haptic Interfacing: This allows for users to interact with these applications physically by manipulating the device's shape in a specific fashion. Below we demonstrate some scenarios for our Media Player and Messenger applications.

Figure 16 shows the control sequence for the Media Player skipping to the next audio, image or video file. These control methods are a requirement during HUD mode operations due to the hidden AR controls. Here the user would (a) grab the device above the base marker, and (b) rotate counter-clockwise until the device reacts to the motion and (c) automatically returns to it's original position. Our robot control and integration software then forwards the registered movement to the AR application and the next file is rendered in the player. The same process is followed to revert to a previous file, the only difference being the opposite (clockwise) rotation of the device by the user.

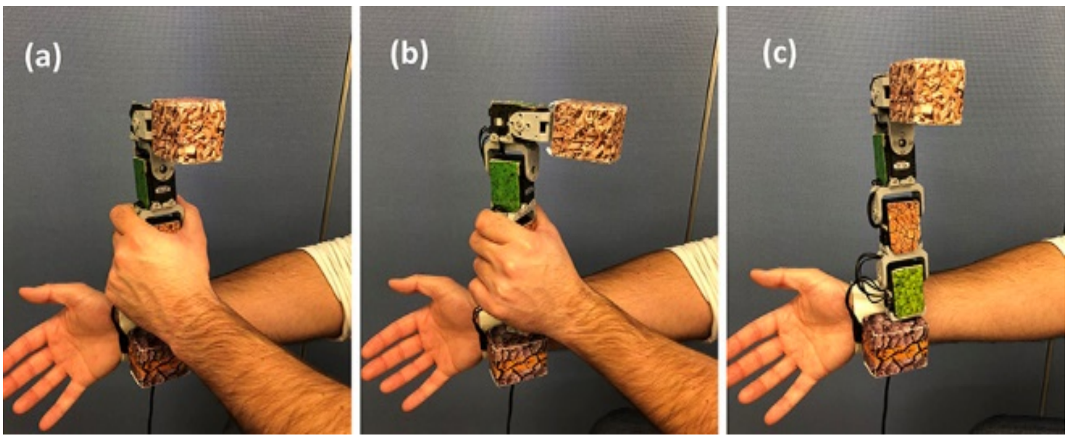

Figure 16 : Media Player control sequence for Previous/Next scenario.

Figure 17 shows the control sequence for the pausing and resuming of media. The user would (a) apply pressure to the device's top marker, and (b) rotate downwards until the device reacts to the motion and (c) automatically returns to it's original position. As above, our software forwards the instruction to the AR application and the media is paused (or resumed if already paused).

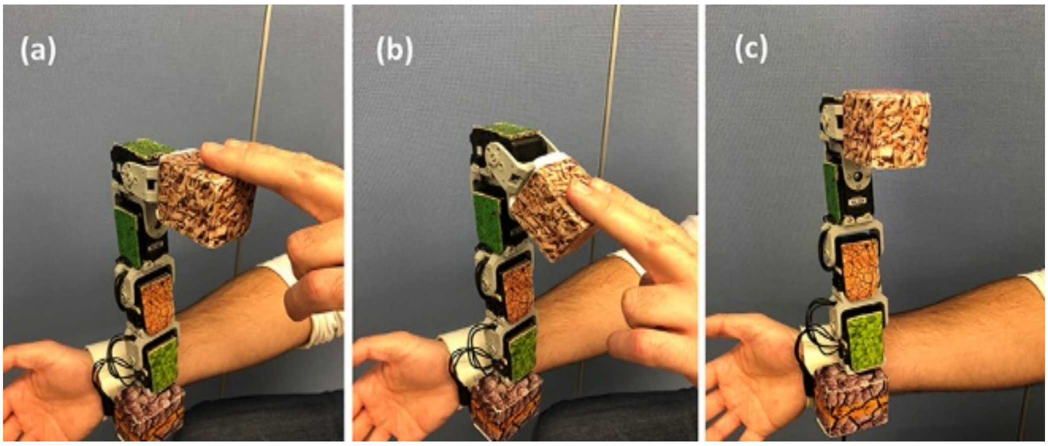

Figure 17 : Media Player control sequence for Play/Pause and Messenger control for Send.

Figure 18 shows the control sequence for sending composed messages within the Messenger. The user would (a) grasp the device's top markers, and (b) rotate downwards until the device reacts to the motion and (c) automatically returns to it's original position. As above, our software forwards the instruction to the AR application and the compiled message in the Messenger input box is sent.

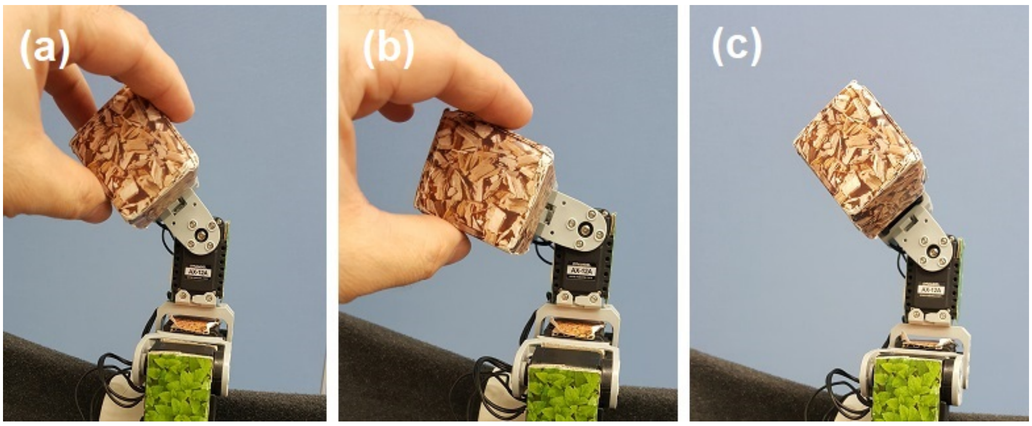

Figure 18 : Media Player control sequence for Play/Pause and Messenger control for Send.

Section 5.6 SRL MODE

The SRL Mode application enables the user to interact with their environment via the wearable device (Figure 19). This application, more than any other, is highly design dependent. The design would influence or even define the capability and functionality of the device in SRL Mode. Different designs could allow for the device to be used as an additional arm, leg, finger or even a tail.

Interactions can thus range from any generic action in place of a user's natural limbs, to interacting with objects or in situations with which a user would otherwise be unable handle without assistance or specialized tools.

Generic interactions, such as handling bottles, opening doors, carrying bags, or typing on a keyboard, free up the user's natural limbs for use on other operations that may require more dexterity than the current device may provide.

More advanced interactions, which may be difficult or possibly dangerous for normal human interaction, such as the handling of objects that are larger than the user could manage, handling objects of extreme temperature, or allowing for single-handed operations usually requiring both hands.

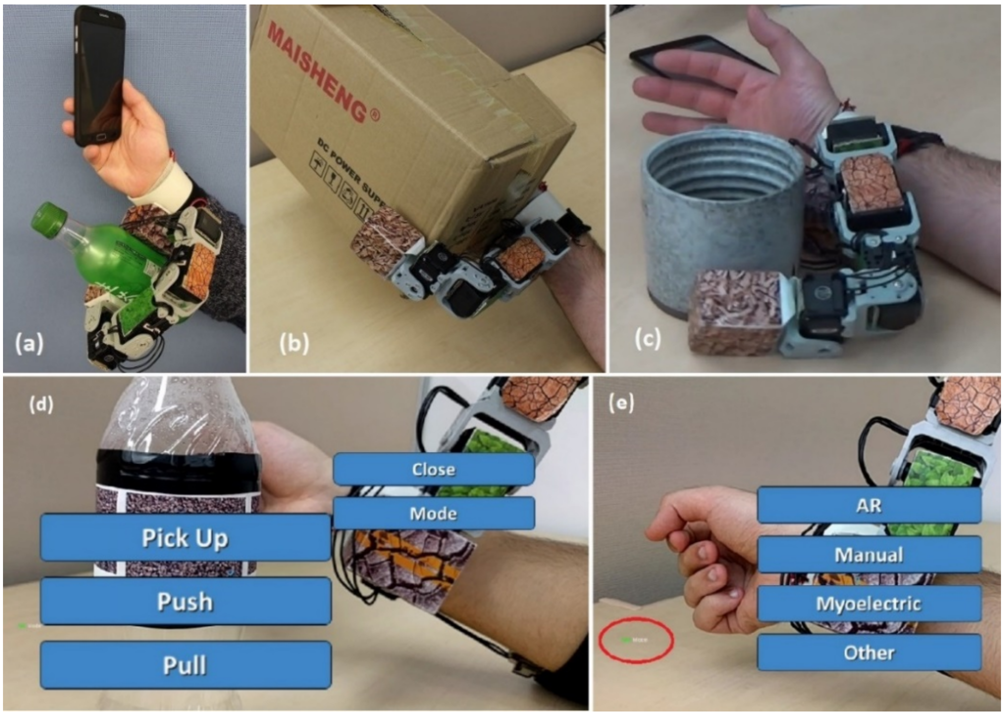

Figure 19 : Examples of SRL mode interactions.

Section 5.7 CONTROLLER

The Controller application extends from the previously mentioned Digital Content and Haptic Interfacing, though it predominantly focuses on the Haptic Interfacing portion. This technology, more commonly referred to as 'Always Available Interfacing', utilizes the wearable device as a type of shape-changing input device or 'Joystick'. This application provides the haptic interfacing for some digital content, whether it be software embedded within the AR interface or running on another 3rd party system connected via the network.

Once launched, the device assumes the 'Joystick' pose (Figure 20). To interact, the user simply morphs the device, the movements of which are calculated and relayed to the designated software as controller input. Conversely, the software is also able to provide force feedback information to device, which is relayed as haptic feedback to the user.

The Controller could be used in a variety of real-world applications, from interfacing with another application within the AR system or controlling a cursor, or game, on a connected PC, to interacting with elements and objects in smart environments.

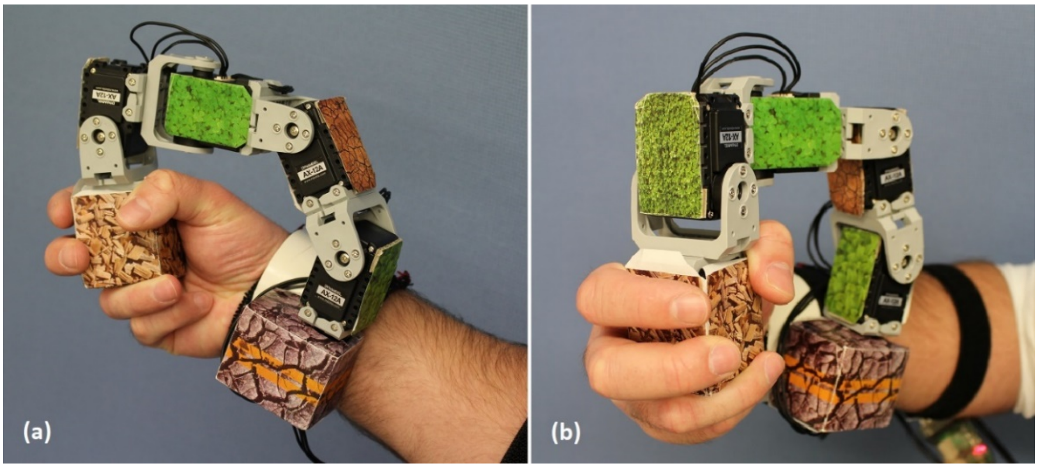

Figure 20 : Device assuming 'Joystick' pose, allowing user to easily morph it's shape.

Section 5.8 AGENT

The Agent application is a form of assistant, and can provide similar functionality to current digital assistants like Siri [47], Cortana [48], or Google Assistant [49], whereby the user may interact with the agent to accomplish tasks and receive information. However, a key difference between our Agent and other digital assistants is the introduction of a physical form.

The Agent application also extends from the above Digital Content and Haptic Interfacing as it also allows for the user to interact with the Agent via both AR and physical interactions. The Agent profile consists of an animated 3-D character model which is rendered over the wearable device. The model and robot movements are synchronized (Figure 21), with interactions directed at one form also affecting the other. These combined factors have an anthropomorphic effect and promote the impression that the user is interacting with another physical entity.

Figure 21 : The synchronized movement of 3-D model and robot performing 'Sit' operation.

The Agent responds to touch and gaze controls and is able to relay any system notifications and feedback to the user in haptic, audio and display format. Some examples of physical interactions that our Agent can provide include system-triggered actions, such as the welcome greeting, whereby the Agent waves and verbally greets the user when the application is started, or user-triggered actions, such as the shock response, whereby the agent flinches and winces as a reaction to being bumped. The Agent is also able to provide feedback for other applications within our implementation, such as message notifications, whereby the Agent taps the user's arm and verbally informs the user that a message was received in the messenger application.

CHAPTER 6. EVALUATION

A. Objectives and Participants

We investigated the user impression of, and usability of, our developed applications and implementation platform. A diverse collection of students between the ages of 21 and 29 were recruited to participate in our study. All participants indicated that they were familiar with AR through common smartphone applications, while most knew of wearable robots through science fiction media or other research.

B. Flow

The evaluation began with a brief introduction of AR and our robot. The participants were walked through the expected applications, they then had the opportunity to experience each for themselves. After each application, users were asked to complete a short survey on their experience. Upon completion, the participants answered a questionnaire including a semi-structured interview (5-point Likert scale, 5 represents strongly agree). The user study lasted 60 minutes per participant and the configuration of our study is as shown in Figure 22.

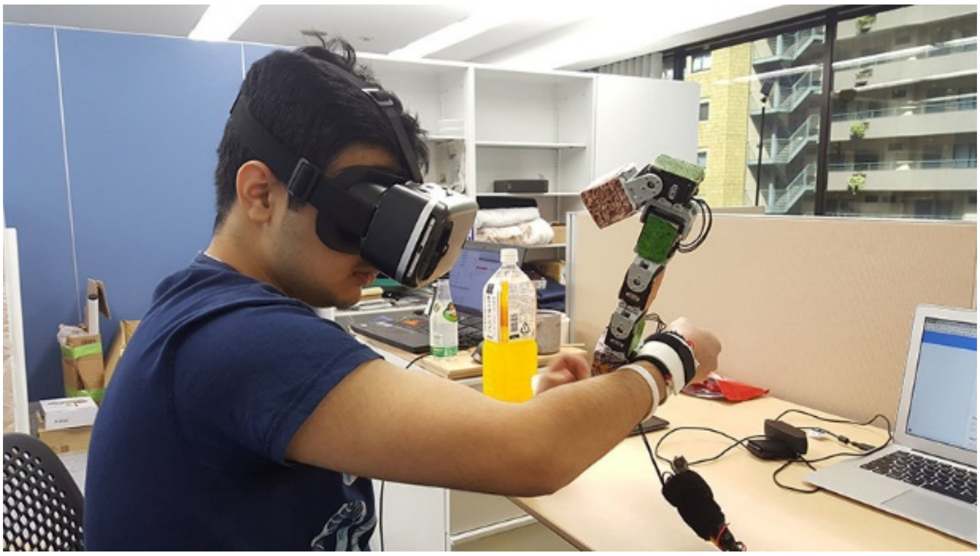

Figure 22 : A participant during the user study as he experiences AR interactions with the wearable robot.

C. Results and Analysis

User Interface: Participants found the interface easy to use (*m*=4.4, *sd*=0.55). Participants indicated that the interactions within our system are novel and enjoyable, and all participants agreed that putting the device into stand-by mode was the best course when not interacting with the system.

Shape-Changing Menus: Participants had mixed feedback for this application. Most participants were able to accurately correlate robot pose and menu state when the AR menus became hidden (*m*=4.4, *sd*=0.89). Participants indicated that the movement when changing robot pose was too subtle when navigating the AR menu. Participants recommend that movement should be more noticeable. We conclude that more distinct poses and faster pose transitions could contribute to such shortcomings.

Device Status: Participant feedback was generally positive about this application (*m*=4.2, *sd*=0.44). They suggested reducing the information to that which could be more useful to an end user, such as temperature and battery readings or failure notifications. A few participants preferred using the HUD display for this application, stating difficulties experienced when tracking between fiducial markers. We further discuss this shortcoming within the next section.

Workspace: Participants feedback was decidedly positive about this application (*m*=5, *sd*=0). Some participants suggested adding a pop-up notification to state when they are too near the device to properly view the workspace advisory. One participant even suggested to extend the application to give the workspace of the user's entire body.

SRL: Participants feedback was positive about this application (*m*=4.4, *sd*=0.55). They shared many possible use cases for the application in their day to day lives, the most resounding of which was simply to carry objects so as to free up their natural limbs for other tasks. Other more advanced tasks would require alterations or adaptations to the current design.

Controller: Participants feedback was highly positive about this application (*m*=4.6, *sd*=0.55). They gave numerous examples of what they would like to use the 'Always Available Interfacing' technology for in the future, such as for a PC or gaming controller or for smart home and appliance control. Participants also suggested adding buttons or sensors to accommodate growing use cases.

Agent: Participants found this application novel, showing great approval of the physical agent form (*m*=4.8, *sd*=0.45). They stated that the interactions with the Agent were enjoyable, but could be improved with the addition of voice commands. There was also a suggestion for using the Agent to execute and operate the system, and other applications, with the user.

This application was ranked most useful for day-to-day life, as shown in Figure 23.

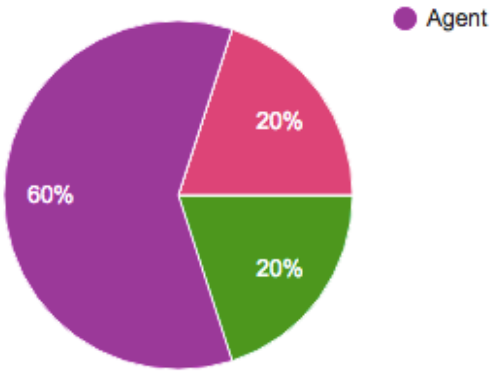

Figure 23 : User results for 'most useful application for daily life'.

Media Player & Messenger: Participants found these applications highly enjoyable (*m*=4.4, *sd*=0.55), admiring the dual control methods for the content on display (*m*=4.8, *sd*=0.45). They also approved of the HUD capable content for use in conjunction with other tasks. User's also approved of additional possible applications listing all their favourite smartphone applications as suggestions, SNS applications holding the most favour.

Robot: Participants stated that the robot was quite comfortable (*m*=3.4, *sd*=0.89) but mentioned that available tasks could benefit from a slightly smaller form factor, this would also address their concerns on the devices weight when used over extended periods of time (*m*=2.4, *sd*=0.55).

AR HMD: Participants stated that the HMD was also quite comfortable (*m*=3.0, *sd*=1.0), but gave mixed reviews on AR performance (*m*=3.4, *sd*=1.14). They mention that AR navigation was easy and intuitive (*m*=4.2, *sd*=0.69), stating approval of our "hands-free" control.

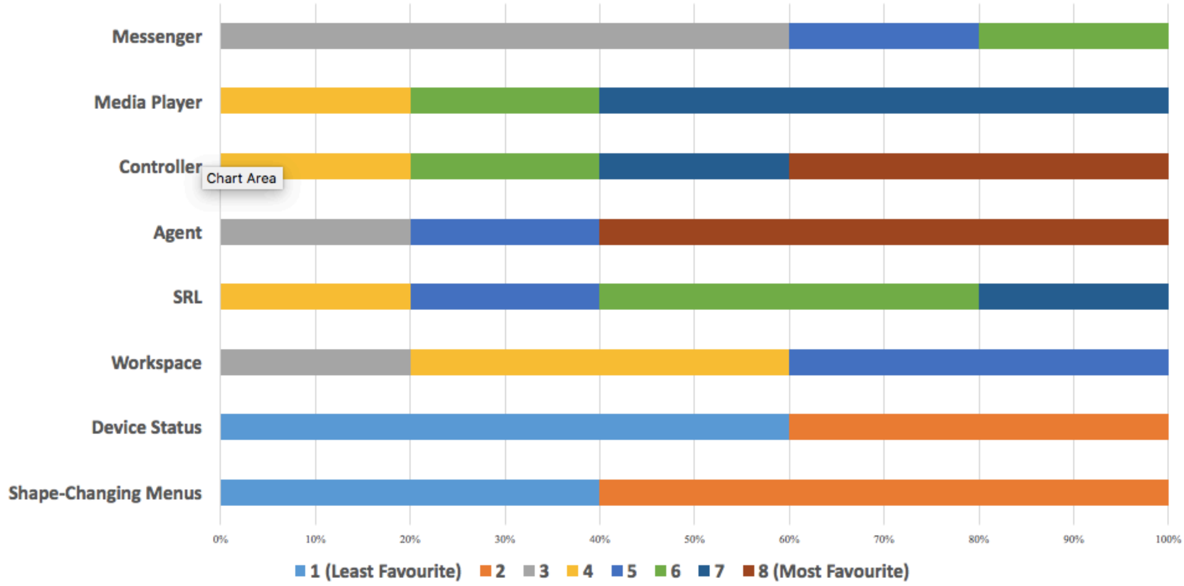

Figure 24 : User results for 'favourite experience of the survey'.

Overall, participants were very satisfied with our approach (*m*=4.8, *sd*=0.45). As shown in Figure 24, The Agent and Controller applications were most enjoyed during the user survey. Participants suggested that including further interaction modalities, like voice commands or eye-tracking, could provide intriguing multimodal interaction capabilities. We believe the results were generally encouraging to pursue further works.

CHAPTER 7. LIMITATIONS AND FUTURE WORK

The implementation of our solution was revealed to have a number of shortcomings within our approach. Our chosen AR hardware, despite its ease of use, has shown some minor issues. The performance of the fiducial marker tracking method varied with differences in lighting and camera focus. This affects the user experience, forcing the user to compensate by periodically moving into closer proximity of the markers. Furthermore, our somewhat obtrusive AR/VR enclosure, and selection thereof, was limited by the technology available at the time. Prolonged use of the resource-intensive application resulted in poor smartphone performance, this included severe battery drain and over-heating. This challenges our vision of an 'always-on' system. Lastly, WiFi is vulnerable to communication delays, which may occur depending on network traffic and degrade the user experience.

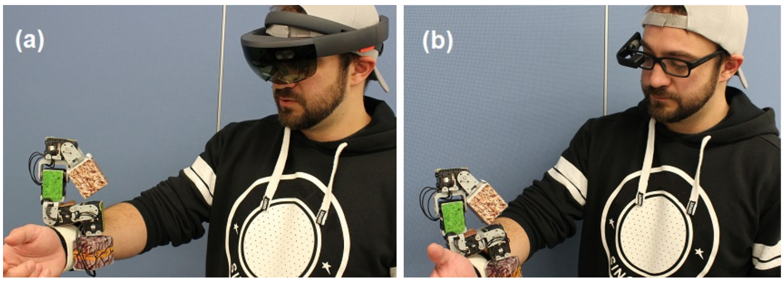

Figure 25 : Examples of (a) alternative and (b) future-type headset possibilities.

To overcome tracking challenges, we intend to extend our evaluation to smart eyewear, such as HoloLens [50] (Figure 25(a)), which have optimized battery life, tracking-performance, connectivity and multiple interaction modalities. Advancements in tracking technologies, such as model tracking [51] or simultaneous localization and mapping [52], are promising and could yield better tracking than our current implementation. In the future, much less obtrusive HMD options may be available, and thus more feasible for an 'always-on' type system, such as the example shown in Figure 25(b). Ad-hoc wireless technologies, such as Bluetooth, could overcome the limitations experienced with WiFi. We also intend to incorporate advanced voice controls to enhance the usability and user experience.

Our framework could also be extended to accommodate further AR and robot enhancements. Advancements in gesture recognition and control systems would allow for the user to interact with a 3-D model of the device in AR, which directly manipulates the physical robot device. Intelligent motion planning systems and sensors would also enable advanced capabilities, such as automated collision avoidance or advanced tele-operation. Figure 26 shows how this same device could even be used when detached from the user, where AR allows the user to continue observation and maintain control of the device even when the device is out of sight. The device could be controlled by a remote guest, allowing for the device to act as proxy for that guest's physical being, and the agent as the guest's avatar. Such advancements would require some improvements to the services offered within our future framework.

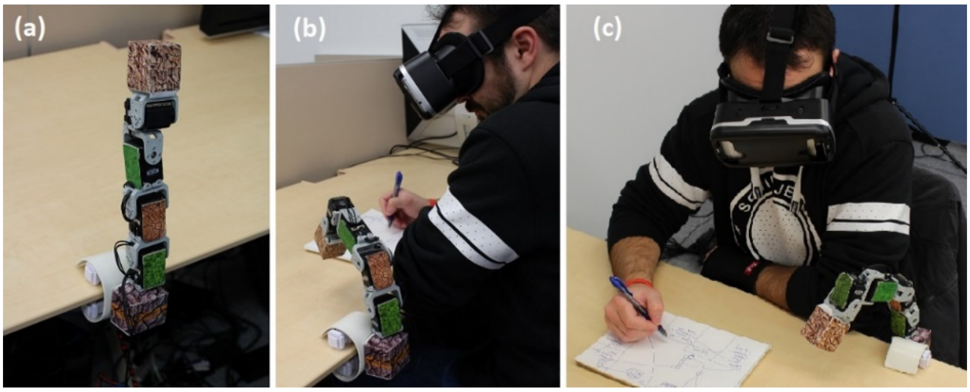

Figure 26 : Examples of detached interaction.

CHAPTER 8. CONCLUSION

In the future, daily worn robots will play a larger role in our lives. These wearables have inherent control and feedback difficulties, especially as they lack embedded input methods that are suitable across a variety of contexts. Therefore, we bridge this gap with a framework that enables the development of applications and experiences that combine AR and wearable robots. After describing the details of our design space and framework, we developed a solution with several user experiences and applications demonstrating the feasibility of our framework.

We proceeded with an evaluation of these experiences. Results indicated that users enjoyed the agent application and haptic interaction using the robot, but stated the device status information would be better suited to technicians or professionals requiring such information provided. User feedback, and overall survey results, encourage us to pursue further work, specifically addressing technical challenges in tracking and networking, as well as exploring the multimodal interaction potential of our robot in different usage scenarios.

REFERENCES

[1] S. Leigh and P. Maes, "Body Integrated Programmable Joints Interface," Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16), pp. 6053-6057, 2016.
[2] A. Dementyev, et al, "Rovables: Miniature On-Body Robots as Mobile Wearables," Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16), pp. 111-120, 2016.
[3] H. Kao, et al, "Exploring Interactions and Perceptions of Kinetic Wearables," Proceedings of the 2017 Conference on Designing Interactive Systems (DIS '17), pp. 391-396, 2017.
[4] M. Kolsch, R. Bane, T. Hollerer and M. Turk, "Multimodal interaction with a wearable augmented reality system," IEEE Computer Graphics and Applications, vol. 26, no. 3, pp. 62-71, 2006.
[5] M. Lee, M. Billinghurst, W. Baek, R. Green, and W. Woo, "A usability study of multimodal input in an augmented reality environment," Virtual Reality, vol. 17, no. 4, pp. 293-305, 2013.
[6] S. Henderson and S. Feiner, "Exploring the Benefits of Augmented Reality Documentation for Maintenance and Repair," IEEE Transactions on Visualization and Computer Graphics, vol. 17, no. 10, pp. 1355-1368, 2011.
[7] B. Schwald and B. De Laval, "An augmented reality system for training and assistance to maintenance in the industrial context," Journal of WSCG, vol. 11, no. 3, pp. 425-432, 2003.
[8] V. Heun, J. Hobin, and P. Maes, "Reality Editor: Programming Smarter Objects," Proceedings of the 2013 ACM conference on Pervasive and Ubiquitous Computing adjunct publication (UbiComp '13 Adjunct), pp. 307-310, 2013.
[9] V. Heun, S. Kasahara, and P. Maes, "Smarter Objects: Using AR Technology to Program Physical Objects and their Interactions,". CHI 2013 Extended Abstracts on Human Factors in Computing Systems (CHI EA '13), pp. 961-966, 2013.
[10] M. Walker, H. Hedayati, J. Lee, and D. Szafir, "Communicating Robot Motion Intent with Augmented Reality," Proceedings of 2018 ACM/IEEE International Conference on Human-Robot Interaction (HRI '18), pp. 316-324, 2018.
[11] E. Rosen, et al, "Communicating Robot Arm Motion Intent Through Mixed Reality Head-mounted Displays," arXiv e-print, arXiv:1708.03655, 2017.
[12] R. Chadalavada, H. Andreasson, R. Krug, and A. Lilienthal, "That's on my Mind! Robot to Human Intention Communication through on-board Projection on Shared Floor Space," 2015 European Conference on Mobile Robots (ECMR), pp. 1-6, 2015.
[13] N. Tran, J. Rands, and T. Williams, "A Hands-Free Virtual-Reality Teleoperation Interface for Wizard-of-Oz Control," Proceedings of the International Workshop on Virtual, Augmented, and Mixed Reality for HRI, 2018.
[14] S. Hashimoto, A. Ishida, M. Inami, and T. Igarashi, "TouchMe: An Augmented Reality Based Remote Robot Manipulation." Proceedings of International Conference on Artificial Reality and Telexistence (ICAT 2011), 2011.
[15] J. Frank, S. Krishnamoorthy, and V. Kapila, "Toward Mobile Mixed-Reality Interaction with Multi-Robot Systems," IEEE Robotics and Automation Letters, vol. 2, no. 4, pp. 1901-1908, 2017.
[16] R. Grieder, J. Alonso-Mora, C. Bloechlinger, R. Siegwart, and P. Beardsley, "Multi-robot control and interaction with a hand-held tablet," In Workshop Proceedings of International Conference on Robotics and Automation (ICRA 2014), vol. 131, 2014.
[17] M. Fiala, "A Robot Control and Augmented Reality Interface for Multiple Robots," Proceedings of Canadian Conference on Computer and Robot Vision, pp. 31-36, 2009.
[18] T. Williams, N. Tran, J. Rands, and N. Dantam, "Augmented, Mixed, and Virtual Reality Enabling of Robot Deixis," Proceedings of the International Conference on Virtual, Augmented, and Mixed Reality, 2018.
[19] T. Holz, M. Dragone, and G. O'Hare, "Where Robots and Virtual Agents Meet," International Journal of Social Robotics, vol. 1, no. 1, pp. 83-93, 2009.

[20] T. Willaims, "A Framework for Robot-Generated Mixed-Reality Deixis," Proceedings of the International Workshop on Virtual, Augmented, and Mixed Reality for HRI, 2018.
[21] Myo.com, "Myo Gesture Control Armband," [Online] https://www.myo.com/ [Accessed: 15 April 2018].
[22] S. Leigh, T. Denton, K. Parekh, W. Peebles, M. Johnson, and P. Maes, "Morphology Extension Kit: A Modular Robotic Platform for Physically Reconfigurable Wearables," Proceedings of the 12th International Conference on Tangible, Embedded, and Embodied Interaction (TEI '18), pp. 11-18, 2018.
[23] T. Sasaki, Y. Saraiji, C. Fernando, K. Minamizawa, and M. Inami, "MetaLimbs: multiple arms interaction metamorphism," Proceedings of SIGGRAPH '17 Emerging Technologies, 2017.
[24] D. Clode, "The Third Thumb," [Online] https://www.daniclodedesign.com/thethirdthumb [Accessed: 21 April 2018].
[25] A. Jackowski, M. Gebhard, and R. Thietje, "Head Motion and Head Gesture-Based Robot Control: A Usability Study," IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 26, no. 1, pp. 161-170, 2017.
[26] E. Faidallah, Y. Hossameldin, S. Rabbo, and Y. El-Mashad, "Control and modeling a robot arm via EMG and flex signals," Proceedings of the 15th International Workshop on Research and Education in Mechatronics (REM '14), pp. 1-8, 2014.
[27] P. Dimitriadis and J. Alexander, "Evaluating the Effectiveness of Physical Shape-change for In-pocket Mobile Device Notifications," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14), 2014.
[28] E. Hogan, et al, "An Exploration of Mobile Shape-Changing Textures," Proceedings of the Eleventh International Conference on Tangible, Embedded, and Embodied Interaction (TEI '17), pp. 275-282, 2017.
[29] E. Pedersen, S. Subramanian, K. Hornbaek, "Is my phone alive?: a large-scale study of shape change in handheld devices using videos," Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14), pp. 2579-2588, 2014.
[30] F. Hemmert, G. Joost, A. Knorig, and R. Wettach, "Dynamic knobs: shape change as a means of interaction on a mobile phone," Proceedings of Extended Abstracts on Human Factors in Computing Systems (CHI EA '08), pp. 2309-2314, 2008.
[31] Y. Park, J. Park, and T. Nam, "The Trial of Bendi in a Coffeehouse: Use of a Shape-Changing Device for a Tactile-Visual Phone Conversation," Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI '15), pp. 2181-2190, 2015.
[32] K. Hartman, J. McConnell, B. Koutoukov, H. Predko, and I. Colpitts-Campbell, "Monarch: Self-Expression Through Wearable Kinetic Textiles," Proceedings of the 9th International Conference on Tangible, Embedded, and Embodied Interaction (TEI '15), pp. 413-414, 2015.
[33] F. Hemmert, M. Lowe, A. Wohlauf, and G. Joost, "Animate mobiles: proxemically reactive posture actuation as a means of relational interaction with mobile phones," Proceedings of the 7th International Conference on Tangible, Embedded and Embodied Interaction (TEI '13), pp. 267-270, 2013.
[34] P. Strohmeier, J. Carrascal, B. Cheng, M. Meban, and R. Vertegaal, "An Evaluation of Shape Changes for Conveying Emotions," Proceedings of CHI Conference on Human Factors in Computing Systems (CHI '16), pp. 3781-3792, 2016.
[35] K. Nakagaki, S. Follmer, and H. Ishii, "LineFORM: Actuated Curve Interfaces for Display, Interaction, and Constraint," Proceedings of the 28th Annual Symposium on User Interface Software and Technology (UIST '15), pp. 333-339, 2015.
[36] K. Nakagaki, A. Dementyev, S. Follmer, J. Paradiso, and H. Ishii, "ChainFORM: A Linear Integrated Modular Hardware System for Shape Changing Interfaces," Proceedings of the 29th Annual Symposium on User Interface Software and Technology (UIST '16), pp. 87-96, 2016.
[37] Unity3D.com, "Unity3D," [Online] https://unity3d.com/ [Accessed: 28 March 2018].
[38] Vuforia.com, "Vuforia in Unity3D," [Online] https://www.vuforia.com/ [Accessed: 15 April 2018].
[39] Robotis.com, "ROBOTIS," [Online] http://www.robotis.com/ [Accessed: 21 January 2018].
[40] GDP.hk, "GPD WIN," [Online] http://www.gpd.hk/gpdwin.asp [Accessed: 21 March 2018].

[41] I. Fette, “The websocket protocol,” 2011, [Online] https://tools.ietf.org/html/rfc6455 [Accessed: 15 April 2018].
[42] JSON.org, “Introducing JSON,” [Online] https://www.json.org/ [Accessed: 15 April 2018].
[43] J. Kaiser and M. Mock, “Implementing the real-time publisher/subscriber model on the controller area network (CAN),” Proceedings of 1999 IEEE International Symposium on Object-Oriented Real-Time Distributed Computing (ISORC '99), pp. 172-181. 1999.
[44] M. Matteucci, “Publish/Subscribe Middleware for Robotics: Requirements and State of the Art,” Technical Report, 2003.
[45] N. Andersson, et al, “AR-Enhanced Human-Robot-Interaction - Methodologies, Algorithms, Tools”. Procedia CIRP, vol. 44, pp. 193-198, 2016.
[46] ROS.org, “ROS,” [Online] http://www.ros.org/ [Accessed: 18 April 2018].
[47] Apple.com, “Siri,” [Online] https://www.apple.com/ios/siri/ [Accessed: 8 June 2018].
[48] Microsoft.com, “Cortana,” [Online] https://www.microsoft.com/en-us/cortana [Accessed: 8 June 2018].
[49] Google.com, “Google Assistant,” [Online] https://assistant.google.com/#?modal_active=none [Accessed: 8 June 2018].
[50] Microsoft.com, “Detail of light reflection on lens of HoloLens,” [Online] https://www.microsoft.com/en-us/hololens [Accessed: 16 April 2018].
[51] Vuforia.com, “Model Targets – 2.0 API Overview,” [Online] https://library.vuforia.com/articles/Solution/model-targets-user-guide.html [Accessed: 16 April 2018].
[52] Wikitude.com, “SLAM (Simultaneous Localization & Mapping) Augmented Reality SDK That Enables Markerless and Marker-based AR Projects,” [Online] https://www.wikitude.com/wikitude-slam/ [Accessed: 15 April 2018].