DEPARTMENT OF INFORMATICS

Technische Universität München

Bachelor’s Thesis in Informatics: Games Engineering

Augmented and Virtual Reality User Interface with a Virtual Embodied Controller for Object Manipulation

Jan Gerling

DEPARTMENT OF INFORMATICS

Technische Universität München

Bachelor's Thesis in Informatics: Games Engineering

Augmented und Virtual Reality User Interface mit einem virtuell verkörperten Controller zur Objektmanipulation

Augmented and Virtual Reality User Interface with a Virtual Embodied Controller for Object Manipulation

Author:	Jan Gerling
Supervisor:	Prof. Dr. Gudrun Klinker
Advisor:	M.Sc. Sandro Weber
Submission Date:	15.01.2018

I confirm that this bachelor’s thesis is my own work and I have documented all sources and material used.

Garching, 15.01.2018 Jan Gerling

Abstract:

In this Bachelor thesis, a user interface for the manipulation of virtual 3D-Objects in an augmented or virtual reality environment was developed. The user interface is controlled with gestural commands by measuring the electric potential emitted by the user's muscles and rotation of the forearm with a Myo armband. With the proposed user interface, the user can move and rotate virtual objects to perform multiple complex tasks, e.g. building a tower with cubes. The user interface utilizes virtual embodiment to increase performance and facilitate natural behavior. By utilizing the Myo armband for input, the user has high mobility and a high degree of freedom.

A quick evaluation of the Myo's performance, with focus on gesture classification and spatial and rotational tracking showed, that the Myo's rotational tracking is good. However, the average correct gesture detection rate of 66% is very low. Furthermore, the device cannot be used for absolute spatial tracking of the wearer's arm. Implementing sophisticated haptic feedback is not possible with the current MyoSDK.

To evaluate the user interface a user study with 18 participants was conducted. The evaluation of the user interface regarding performance and usability showed that the proposed user interface is perceived as good by the users with an average SUS score of 81. Also, the users can perform complex tasks. Applying the armband's rotation to control a pointer is very efficient. The users can focus small objects within a distance of five meters accurately and fast. The user study showed that the proposed user interface is self-explanatory and intuitive to use. Nonetheless, further improvements on feedback, gesture classification, and precision are necessary. The user interface is lacking efficiency due to limited control and insufficient feedback.

Acknowledgements

I would like to thank Professor Gudrun Klinker and the research group Augmented Reality (FAR) of the Chair of Computer Aided Medical Procedures and Augmented Reality (CAMPAR) at the Technical University of Munich (TUM).

Very warmly I would like to thank my advisor Sandro Weber for offering me the great chance to write this bachelor thesis. I am very thankful for your constant advise and support.

Furthermore, a big thank to Sarah Hanus, Michael Hanus, Anne Gerling and Reinhard Gerling for reviewing my thesis.

I would like to thank all the people whom took part in the user study and the Myo evaluation.

Table of Contents

List of abbreviations

API:	application interface
AR:	augmented reality
EMG:	electromyogram
HDC:	hyperdimensional computing
HMD:	head-mounted display
IMU:	inertial measurement unit
SDK:	software development kit
SUS:	system usability scale
SVM:	support-vector machine
UI:	user interface
VR:	virtual reality

List of figures

List of tables

1 Introduction

The subject of this thesis is introduced in this chapter. Furthermore, the course of major activities is explained.

1.1 Motivation

In recent years the fields of virtual reality (VR) and augmented reality (AR) made vast progress. The visual presentation has reached unseen quality, the technologies have dropped in price and are now available to a huge consumer market, e.g. the HTC Vive virtual reality setup is available for only 700€ [1]. Also, smartphones, consoles, PCs etc. have turned into gateways to immersive environments offering the user way more than the old 2D-Screen technologies. Many major companies like Facebook, Unity-Technologies, Microsoft, Google, Apple, and HTC are funding and spearheading this development with large investments [2].

With these new technologies, many new types of use have emerged and many more are yet to be discovered. Engineers using augmented reality to test and improve their concepts, doctors use these technologies to help their patients, and managers can monitor the current workload of their employees in a factory accurately [3].

Microsoft has driven the development for augmented reality in recent years with the Microsoft HoloLens, a Mixed-Reality-Smartglass projecting and anchoring three-dimensional objects into the real world as seen by the user. The user can interact with this augmented reality environment using gestures, voice commands and gaze [4]. Unfortunately, the user can only interact within the HoloLens’s field of view, which is very limited, interacting with objects outside the field of view is not possible. This limitation is heavily influencing the immersion of the users and their ability for natural interaction.

The progress made for gesture recognition, as deployed by the Microsoft HoloLens, can be combined with another field of scientific research, the electromyography (EMG). EMG based gestural interfaces are new and currently only slightly explored, but they hold great opportunities for new user interfaces. These interfaces are highly mobile, because they do not require stationary sensors like cameras. The user input can be identified and classified reliable, unlike other techniques like voice commands and the user can interact intuitively with the interface. Another great advantage is that the interface can be extended or customized by adding or replacing gestures. One device using electromyography to accomplish all the named advantages is the Myo by Thalmic Labs [5].

The goal of this thesis is to develop a user interface for an augmented reality allowing the user to manipulate virtual objects. The user is supposed to exceed the limitations of the used head-mounted display (HMD), particularly the limitation of action to the camera and user field of view. Using the Myo armband and virtual embodiment the user is supposed to naturally interact with virtual objects, even if the user cannot see these objects. For example, if the objects are behind the user’s back or sight is limited due to other obstacles, like tables.

The sense of embodiment with the user interface is critical to empower the user to overcome these obstacles. With a high sense of embodiment, the user can add the virtual controller to his body schema and thereby improve his efficiency with the controller [6], [7]. Furthermore, the user is empowered to control the virtual body, without continuous oversight. The user can map the position and other properties of the controller with his or her extended body schema [7].

1.2 Course of Activities

To develop the user interface using the Myo armband, the key functions of the armband were evaluated to design the interface according to Myo’s strengths and weaknesses. Unfortunately, there is only few research data or valuable information available concerning the gesture classification performance and spatial and rotational tracking of the Myo. Therefore, a quick experiment to assert the devices performance was necessary.

Afterwards, the user interface was developed and implemented in an iterative process. This process was chosen to avoid design flaws, keep the interface simple and self-explanatory and to review the current state of development. The user interface was designed and implemented in small steps. These steps were frequently tested with different users, regarding the before mentioned issues.

To evaluate the designed user interface, a user study was conducted. The testers were solving multiple tasks and rated the user interface with a system usability scale questionnaire. Further data to task performance, like time, reported issues, and success was collected.

2 Related Research

This chapter summarizes the research done to understand the theme. Its main topics are virtual embodiment, electromyography, and gesture classification.

2.1 Definition and Description of Virtual Embodiment

People experiencing a virtual or augmented reality feel a subjective presence in this reality. The term presence is described as “the match between proprioception and sensory data” [8][p.3] and “the extent to which the information displayed allows individuals to construct their own internal mental models of reality” [8][p.3]. To further increase this sensation consistent tracking, movement and displaying should be given at all time [8]. This sense of presence enables the user to act and react as if he or she was in the virtual reality [9].

The perception of being in a different place can be extended to the illusion of having a virtual body. This is called virtual embodiment [9]. This illusion is not only restricted to realistic bodies but also unrealistic ones, like having extremely long arms or an artificial hand. They allow the user to perform actions or have experiences not possible with their real body [6], [10]. These experiences are called out of body experiences and heavily rely on a sense of embodiment for the virtual body. Sense of embodiment occurs if the, “bodies [Note] properties are processed if they were properties of one’s own biological body.” [11][p.375]. Furthermore, virtual embodiment consists of three main properties, sense of self-location, sense of agency and sense of body-ownership. The sense of self-location describes the person’s spatial experience of being inside the body [11]. The sense of agency is the person’s experience of controlling the body with actions and intentions and the sense of body-ownership describes the experience of owning the virtual body [11].

Research shows that people with a virtual body are more likely to behave in a natural way within virtual environments, by avoiding risks [12] and performing actions according to their virtual body capabilities [6].

2.2 Basics of Electromyography and Gesture Classification

Every device evaluating and measuring the electrical potential generated by the skeletal muscles is called an electromyograph [13]. The resulting recorded data is called an electromyogram. The skeletal muscles emit electricity when they are used for example, while forming a fist with the hand. This electricity can be measured as EMG data. Furthermore, the muscles respond to stimulation from nerves induced by neurological activity. Therefore, also patients with amputations can use electromyography, for example to control a prosthesis.

Gathering electromyography data can be separated in two types, invasive and non-invasive. Invasive electromyography might be used in medical environments, usually needles are inserted in the muscle and then the EMG data is collected. This technique requires medical personal like a physician and special equipment [14]. The other type is surface electromyography, for this technique sensors placed on the skin are collecting the EMG data [14]. These sensors are often linear arrays of electrodes measuring the electric potential at multiple points in a line [15]. This technique is easy to use and requires no special personal, but the data might be influenced by a multitude of factors, like body fat, temperate, skin coverage, humidity, hydration and hair [16]. Especially body fat is an important factor, because with increasing distance to the muscles the EMG records have a higher variance [17]. The EMG signal can vary from 0 – 10 mV with a frequency ranging from 0 to 1000 Hz [18]. The most interesting EMG signals are in a frequency ranges from 15 to 25 Hz and 400 to 500 Hz. These frequencies are best for high-pass and low-pass filtering [18].

To classify gestures from EMG data, a multitude of approaches is currently deployed. First, the EMG data is preprocessed by segmentation, extraction of features and reduction of dimension and feature space [19]. Especially spatial segmentation and filtering has proven to be very effective in increasing the classification accuracy [20]. A common approach to classify gestures is to use support-vector machines (SVM) [21], [22]. Also, neural networks, k-means clustering or linear discriminant analysis are used for classification [19][p.1], [22], [23]. Some of these classifiers are also combined e.g. SVMs and neural networks [21]. A recently released approach uses hyperdimensional computing to classify electromyography signals with a high accuracy, ~97.8%, and small training data compared to SVMs [24]. Also, it is very robust and requires only low energy and is therefore well suited for mobile or wearable devices [25]. A multitude of research shows, that classification with an accuracy rate of 85% [26] or even greater than 90% [26], [24], [22] is possible.

3 Introduction to Myo

The Myo armband was developed and produced by the Canadian company “Thalmic Labs” and first sold in the United States of America in 2015 [27]. The Myo armband measures the electrical potential generated by the muscles and maps this data with known patterns for a set of gestures. Therefore, the Myo can be classified as an electromyograph.

Currently, the Myo is trained to recognize five hand gestures (see gestures chapter). The EMG data is measured at the user’s forearm by eight measuring units at a rate of 200 Hz. To optimize the measurement, the widest area of the forearm should be used. Thus, the coverage of skin by the sensor plates is maximized. The eight sensors are connected by flexible bands (see Figure 1), allowing the armband to be worn by a wide range of people, with a forearm circumference from 19cm - 34cm. With a weight of only 93g the Myo armband is extremely light and, therefore, highly mobile [5]. Furthermore, the Myo armband is collecting the data of a nine-axis inertial measurement unit (IMU) at a 50 Hz rate. The IMU consists of a three-axis magnetometer, a three-axis accelerometer and a three-axis gyroscope. The collected data is constantly streamed to a connected device via Bluetooth [28]. Before usage the device is always warmed up and calibrated by the user for better performance. This process takes about two minutes. During the warm-up and calibration process the device is also detecting whether it is worn on the left or right arm, by comparing the Wave Out gesture. It is possible to wear and use two Myos at the same time, one at each arm [29]. Due to the gyroscope and the magnetometer it is possible to determine the Myo’s rotation accurately, thereby, receiving a relative orientation of the user’s forearm. The data received by the accelerometer can be used to determine the relative movement of the Myo. Unfortunately, the accelerometer is not very accurate and the available options for processing the data would further increase the error [30].

The Myo comes with native support for multiple platforms like Windows, Mac OSX, Linux, iOS and Android [5]. Further, a software development kit (SDK), with the current version 0.9.0, for the common game engine Unity3D is available [31].

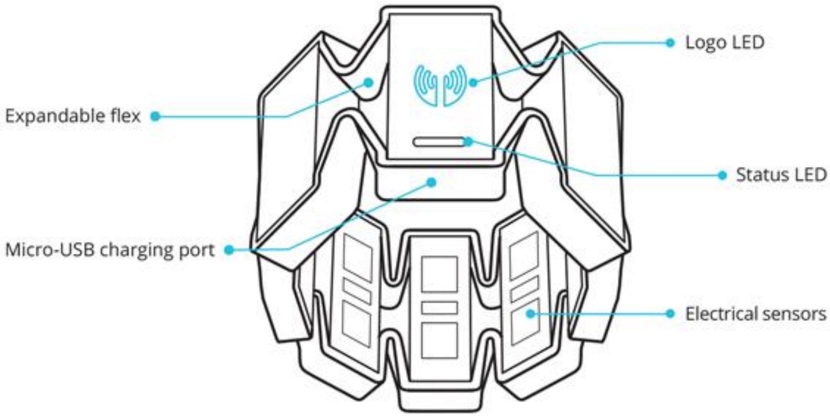

Figure 1: Myo armband [32]

3.1 Why use Myo

The Myo armband is a combined haptic and gestural interface. There are a multitude of advantages the Myo offers for a AR and VR interfaces. First, the user does not need to hold a controller or other device in his or her hands, the hands remain free for other interactions. Especially in augmented reality environments this is very helpful, because the user does have a high degree of freedom in his or her interactions. For example, the user can interact with virtual objects in an augmented reality and moments later use a pencil to write down notes on a paper sheet.

A study found that many users perceive the set of gestures and the general idea very intuitive and easy to understand. The same study also claims that users are very happy with the Myo in a gaming context [33][p. 119], indicating a high potential in other areas of usage.

Also, no additional setup is necessary to use the Myo. It can be worn and used in many environments [33][p.119]. This is a huge advantage over outside-in tracking techniques, e.g. used by the SteamVR or Oculus Rift VR setups. The Myo gives the user huge mobility and little discomfort in using the device.

The Myo provides haptic feedback to the user by vibrating [34]. This can be used to notify the user, e.g. change states, recognized gestures and object collision. This rich pool of possible applications can be very helpful in designing a purposeful user interface, by defining clear feedback patterns which the user can understand quickly.

3.2 Hand Gestures

Currently, the Myo recognizes five distinct hand gestures: Fist, Fingers Spread, Wave Left, Wave Right and Double tap (see figure 3). Also, a rotation and a pan gesture are recognizable, a total of seven gestures. This set of gestures gives the user a wide range to express his or her intentions, e.g. fist gesture for grabbing objects or general affirmative actions.

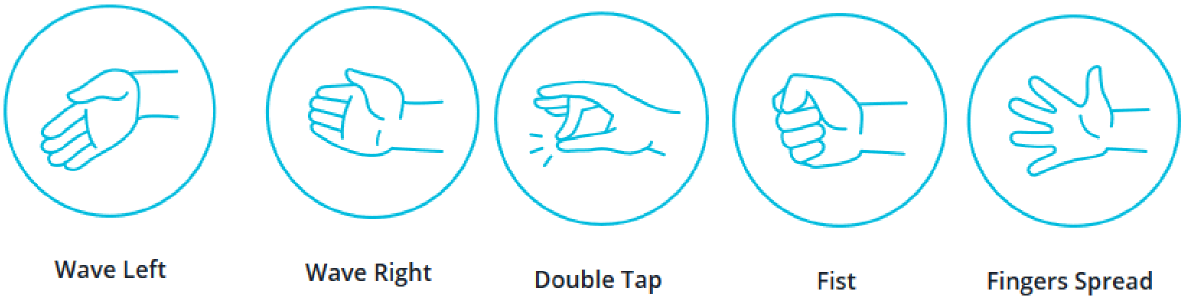

Figure 2: Myo hand gestures [35]

Every gesture is recognized and classified by the Myo device and drivers. Also, these predefined gestures can be combined with arm movements, position and rotation to create many more gestures and meanings [35]. The EMG data is provided in an 8 x 8bit data format and can be used to train on Myo gestures [36].

4 Evaluation of the Myo

The purpose of the evaluation of the Myo armband was to retrieve adequate data to identify the strengths and weakness of the device. For the evaluation of the Myo armband multiple subjects were of special interest: drift and error of rotational tracking, misclassification of gestures, correct classification of gestures and false-negatives for gesture classification. Furthermore, accuracy and drift of the accelerometer to estimate its suitability for the user interface, e.g. for absolute spatial tracking of the user's arm.

Otherwise, developing the user interface was prone to failure, because there is little scientific data available for the Myo. Developing fundamental ideas for the user interface, like choosing gestures for object interaction, or whether to use absolute rotational and spatial tracking or not, would have been based on insufficient assumptions.

4.1 Methodology

To keep the results comparable for the evaluation the standard profile of the Myo was used. The results for a user might not to be optimal by using the standard profile but creating a custom profile is time intensive and often the classification quality varies strongly.

Four persons took part in the evaluation. The testers were of mixed sexes. They were 18 – 29 years old. Only one of the testers had previous experience with the Myo armband. Every tester was wearing a single Myo armband on the widest part of his or her preferred forearm. Before the test, every user warmed up the Myo device for about two minutes and went through the Myo calibration. Furthermore, every user spent a few minutes training the hand gestures multiple times to get familiar with the gestures and icons indicating a certain hand gesture.

For every person, the test took about 30 minutes. The testers were shown an icon on the display and then they had 3.2 seconds to perform this gesture. The gesture to be shown was randomly selected from the five available hand gestures. If the tester performed a gesture, he or she received feedback by vibration of the Myo. If the performed gesture, was wrong or none was performed the set marked as wrong and therefore was not be counted in the statistics. In case the tester performed a gesture, which was not recognized, the tester held the gesture for 3.2 seconds. Afterwards for another 1.5 seconds a rest gesture was show to the tester and he or she was supposed to relax his or her hand. Then a new set of gestures was started. Thus, every test person performed roughly 300 hand gesture during the experiment.

A multitude of data was logged to a JSON file for every test set as described above. The data consisted the expected gesture, as shown to the tester, the start time for performing this gesture, the performed gesture and it's time of detection. The time was logged in the DateTime format as provided by the .Net framework and is, therefore, only accurate to the nearest second.

For orientational tracking, only the user perception of correct tracking was asserted. Measuring the orientation tracking accurately and comparing it with the actual rotation of the device would have required a more complex experiment, exceeding the limitations of this thesis.

To evaluate the rotational tracking, the testers were performing a set of arm rotations and then were asked for their mutual perception of accuracy, delay and drift. The testers could respond with a 0 – 10 Likert scale, where zero indicated not perceived at all and ten meaning full and continuous perception of the item.

4.2 Limits of the Methodology

The test group was rather small and very homogeneous. It is difficult to extrapolate the results for a bigger variety of users, because the test group's variety in body fat and age was only small.

The evaluation of the rotation only reflects the personal perception of the testers and no other data, e.g. reference device rotation, was collected for comparison. Also, the rotational tests were limited only to a few minutes, therefore it is difficult for the testers to notice drift.

Retrieving the time, the Myo needs to classify a hand gesture from the collected data is not possible, because it was not logged, when a user started to perform the required gesture.

4.3 Results

In total 1226 sets of gestures were performed during the test, from which 4 were marked as errors, therefore, a total of 1222 data sets were protocolled (see

Figure **3**).

Only two of the gestures, Wave Right and Double Tap, had correct classification higher than 75%. The Fist and Fingers Spread gestures had a correct classification rate above 60% and the last gesture, Wave Left, of only 34.8% (see

Figure **3**). All the hand gestures, except the Double Tap gesture, were detected multiple times, meaning a hand gesture event was reported by the Myo more than once within a short period of time, although the hand gestures were only performed once. This multi detection was particularly often for Wave Left with a total of 49% and Fist with 22% high above the average of 13% for all hand gestures. Every gesture showed to be frequently misclassified with another gesture, the values are ranging from 7.5% to 17.9%. Especially the Fingers Spread gesture is often misclassified with 17.9% (see

Figure **3**). An in deep analysis of the data shows that the Fingers Spread gesture is often misclassified as Fist gesture. All but the Wave Right gesture are not recognized by the Myo armband sometimes, even though the user performed the gesture correctly.

The testers perceived the rotation around the pitch and yaw axes to be very accurate with medians of 8.75 and 9 close to the maximum of 10. The rotation around the roll axis was perceived less accurate with only a median value of 6 (see Figure 4). The testers barely perceived a delay in rotational tracking of their forearm, which is indicated by the low perception median of one. Also, the testers were not feeling any drift of the rotation (see Figure 5).

All the testers found the vibration feedback for gestures tiring and too strong over a longer period.

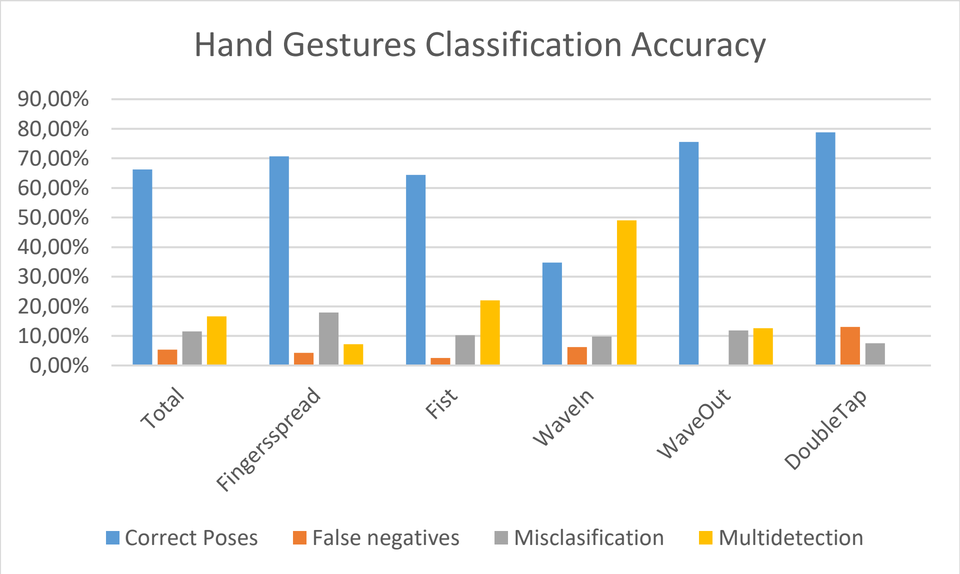

Figure 3: Detection rate in percent for the five hand gestures

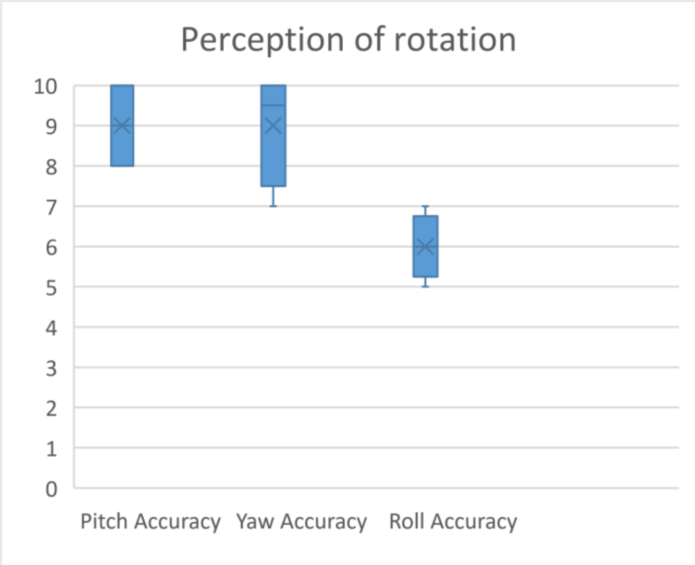

Figure 4: Perception of orientation accuracy in a 0 – 10 Likert scale, higher is better

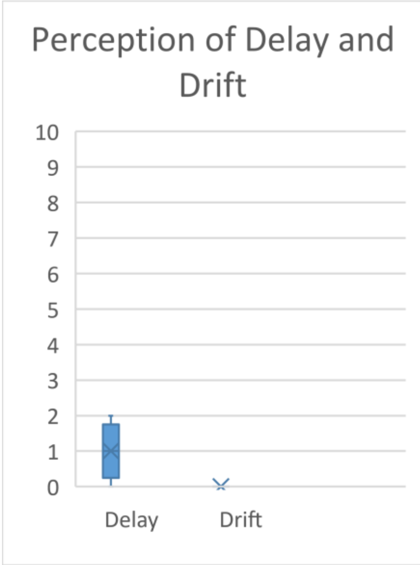

Figure 5: Perception of delay and drift for rotation in a 0 – 10 Likert scale, lower is better

4.4 Discussion of Results

The results show that the gesture classification is inaccurate overall and the reliability of classified gestures varies heavily. The overall classification accuracy of the Myo is much lower than the classification accuracy for other algorithms proposed in research. Thus, the low accuracy shows the classification approach taken by Thalmic Labs requires improvements. Especially, the Wave Left gesture was shown to be very difficult. It is important to point out the correct classification rate for the Wave Right gesture is so low, because during the test the Wave Left gesture was often recognized multiple times (see

Figure **3**). The detection of multiple gestures is a serious problem, because this can result in unintended behavior of the system. An action might be called more times than it was intended to. Furthermore, for every gesture reported, the Rest gesture is reported once by the armband as well, thus it is difficult to distinguish whether or not a user is repeatedly performing a gesture. The results show that the misclassification of gestures is twice as frequent as false-negatives, but the misclassification is also exceeded by the wrong detection of multiple gestures. Therefore, it is most important to find better solutions for the multiple detection and misclassification of hand gestures.

Furthermore, designing the test showed it is not possible with the current SDK v.0.9.0 to define custom haptic feedback patterns. Only the three vibration modes, short, medium and long are accessible to the Bluetooth Application Interface (API) for the Myo. Thalmic Labs released a new Bluetooth header for communication with the Myo armband [37]. This extended communication protocol allows sending more complex vibration commands to the Myo but needs to be implemented and tested.

An explanation of the perceived low accuracy of the roll rotation is as follows. While rotating the forearm, the rotation of the hand is significantly higher than the rotation of the forearm at the elbow joint. Therefore, the Myo detects less rotation at the forearm than at the hand. Also, the degree of rotation of supination and pronation of the human forearm is limited. [38].

The default vibration for the gestures is rather strong and experiencing it over a longer period is tiring for the arm muscles. Therefore, applying the vibration too often or too strongly should be avoided.

Evaluating how quickly a gesture is recognized by the device, might help to compare the suitability of a gesture for a certain task, e.g. throwing. Unfortunately, this is rather difficult to evaluate and was therefore not done. Furthermore, it would have been very important to evaluate the accuracy of the spatial tracking with the three-axis accelerometer of the IMU. Unfortunately, creating a reasonable test proved to be very difficult, because the accelerometer is too inaccurate for absolute tracking [30]. Measuring only the relative movement of the device would have required a more complex experiment as well.

5 User Interface Design

This chapter describes the design of the user interface and all considerations taken into it. The design principles are explained, defined requirements, general approach, feedback mechanics, and further learnings from development process.

5.1 Additional Considerations for Using Myo

The evaluation and the literature review for the Myo device has shown lots of strengths and weaknesses of the Myo armband. The positional tracking lacks accuracy [30] and the vibration feedback does affect the tracking as well. For these reasons, data provided by the accelerometer of the Myo was not utilized, neither for relative or absolute position tracking. Consequently, the interface is always relative to the user camera, e.g. the HoloLens or HMD the user is wearing. The rotational tracking for pitch and yaw axes are accurate (see chapter 4). These axes are used to determine the orientation of the user's arm.

The Wave Left gesture is often detected multiple times and, thereby, creating unintended input. This issued should be avoided by blocking the input selection for a short period of time, within 800 to 1200ms the user should be able to perform a gesture and then release this gesture again. Since most of the hand gestures are having a considerable rate of misclassification, the input is blocked for two seconds.

In general, as few of the hand gestures as possible should be used, because the misclassification of gestures results in unintended actions. To avoid frustration, the Fingers Spread gesture may not be used in the same context as the Fist gesture, because the fist gesture is most likely to be misclassified as Fingers Spread (see chapter 4). Utilizing Fist and Fingers Spread gestures for the same command might be also be effective as well.

5.2 Design Principles

For the user interface, I selected a set of principles by analyzing common user interface design learnings made by usability research [39], [40], [41]. Most of the learnings for user interfaces are based on web, mobile, or desktop applications and differ from virtual reality and augmented reality environments but should be applicable to AR and VR user interfaces as well.

For this work six principles for the UI design were defined:

1. **Keep it simple** – Reduce the complexity of the user interface to a minimum. The user interface is supposed to be self-explanatory, so that the user does not have to learn how to use the user interface. The interface avoids complex contexts, many different gestures or a high variety of interaction options [40].
2. **Virtual embodiment** – Using the Myo device in an environment or circumstances where the user can't see the device requires a certain level of virtual embodiment to be able to use it at all. Furthermore, virtual embodiment will help the user to increase his or her skill in using the interface, resulting in a higher perception of control over the entire system.
3. **Empower the user** – Creating comfort and gaining the reward of feeling skilled with an interface is very important for the user acceptance. The user should be learning and increase his or her control over the interface. This can be achieved by allowing the user to be quicker in his or her actions and be able to increase speed and precision [40].
4. **Consistency –** Most of the users aren't familiar with augmented or virtual realities nor the Myo armband used to control these realities. For these reasons, it is of great importance to keep the interface consistent. Otherwise, the user will take longer to learn patterns for interactions and reach an efficient level of usage. Also, consistency increases the user's comfort with the interface [40].
5. **Communicate what's happening** – Helping the user to understand what's happening is vital for the interface. Therefore, the user is informed whether his or her actions are right or wrong and what the user might do in the current situation.
6. **Be forgiving** – Especially with gestures it is important to tolerate errors, whether they were made by the user or by the system, e.g. by misclassifying of gestures. Also, the user is supposed to learn from failures and avoid them in future actions [40].

Other fundamentals of user interface design like using colors and icons and visual hierarchies are less important, because in the current environment the use of 2D graphical elements, like text fields, is kept at a minimum.

5.3 Requirements

A common approach for user interface design is the use of personas and scenarios [42]. Personas are used to analyze the potential user groups, by creating fictive persons of these groups with defined behavior [43], but for this interface it is difficult to define user groups. Almost all the users won't be familiar with the Myo armband and the Microsoft HoloLens and probably none of them will ever have tried this together. As the user interface is developed and tested in a technical university, the majority of users will be from the age of 18 to 30 with slight experience in virtual reality and high affinity to new technologies. To compensate this lack of information, multiple scenarios were developed (see Appendix 3). A scenario is another tool to model user groups and their behavior in UI design [44].

User stories, an approach from agile development, describe what a user wants to achieve by using a system, e.g. a user interface [45]. By analyzing the user stories (see Table 1), a list of requirements to be fulfilled by the user interface was established.

Requirements:

1. Extend the area of operation of the user to an effective range from 0.2m to 5m
2. High precision to interact with small and distant objects
3. Free movement of the arms in all angles as physically possible for a human
4. Easily enable and disable the user input
5. Distinct action for rotation
6. Distinct actions to grab and drop objects
7. Feedback for gestures and help with objects
8. Release object control at will, to allow the user to throw objects
9. Use physics on all the interactive virtual objects
10. Delete/Destroy objects from the current scene
11. Calibrate the armband at users wish

Solving the selection and handling of multiple objects at once is a complex task. To reduce complexity and avoid extra effort, this function will not be added to the interface.

Table 1: User stories for the user interface based on the case and scenarios

As a	I want to	so that
user	rotate distant virtual objects.	I can see their side even from a distance.
user	grab virtual objects.	I can take a closer look.
user	grab virtual objects.	I can see them from below.
user	grab virtual objects.	I can see them from the above.
user	grab virtual objects from a medium distance (1.5 - 5 m).	To extend my field of operation to a medium sized room.
user	grab virtual objects from a short distance (0.2 - 1 m).	To interact precisely with these objects.
user	retrieve information about objects.	I can distinguish if I can interact with this object.
user	place objects at a certain position.	I can change the environment as I wish.
user	grab two or more objects at once.	I can move sets of objects around.
user	Grab objects out of sight	I can interact with objects, while focusing on another activity.
user	throw virtual objects.	I can interact with the object and the environment physics and thereby experience my control.
user	rotate virtual objects while touching these.	I can see their side.
user	move freely in the environment.	I can reach every reasonable point without limitations.
user	turn on or off the user interface at will.	To control whether I am using the virtual objects or real ones.
user	interact with real objects, alongside virtual objects.	I can use the full freedom of the augmented reality environment.
user	know the current action I can perform.	I can distinguish if I am capable of an action.

Based on the user stories and scenarios, all necessary actions for the user input where synthesized (see Table 2). These actions illustrate the user intentions while using the interface.

Table 2: user actions needed for the user interface

Action	Description
Select	affirmative action, e.g. for grabbing, fetching objects, or selecting menus
Discard	discard all current actions
Discard current action	discard the current action, e.g. dropping object
Menu	disable/enable the UI, recalibrate the armband
Forward	next option in a menu, rotate clockwise

Back	previous option in a menu, rotate counter-clockwise
Fast Forward	increase rotation speed, continuously iterate through options
Fast Back	increase rotation speed, continuously iterate through options

5.4 Proposed User Interface

The user interface design was adjusted to the previously defined requirements and learnings from the Myo evaluation. Furthermore, the UI was frequently adapted to learnings during the development process, see section 5.7.

The UI was designed for a user wearing the Myo armband at the preferred forearm. The position of the Myo representation is relative to the head-mounted display with a fixed offset. The Myo is positioned within the body center, torso, about 60cm below the head. This position does not map with the perceived position of the arm, but it reduces the complexity of the interface, because otherwise the user must consider the rotation of the HMD to correctly map the rotation and position of the Myo armband. This might be improved in future solutions by using body tracking, as supported by the HTC Vive, for positioning the Myo representation instead of the HMD.

The user can rotate the Myo's representation by rotating his or her arm, and then the rotation is directly applied to the model. Direct mapping of the rotation enhances the sense of virtual embodiment with the armband, by granting the user control and body ownership. Research showed that to achieve body self-consciousness, proprioception, body-related visual information, peripersonal space, and embodiment are necessary [46], [47]. The controller is attached to the user's body within the peripersonal space. Due to the synchronous, prolonged, and combined haptic feedback from the Myo (see chapter 5.6) and vision of the controller, the user is embodying the controller [46][p. 152], [48]. The user can constantly map the position of the controller relative to his or her own body position, because the rotation of the user's arm is direct mapped to the controller. Furthermore, the full control leads to a sense of agency and body ownership for the controller [49], [50].

Because the pitch and yaw rotation are accurately tracked by the device (see Myo pretest) tracking errors won't cause relevant discomfort for the user. Furthermore, the user can reset the rotation at any given time.

The arm of the user isn't represented, but the device's forward orientation is displayed by a line (see Figure 6). This line is also representing the pointer. The maximum distance for the pointer is limited to five meters. The range is limited to five meters to fulfill the requirements (see Table 1). A reach of five meters grants the user the ability to reach any point within a medium sized room and ensures high precision on the other hand. The pointer cannot penetrate any virtual objects, it is blocked by virtual walls and objects within five-meter distance (see Figure 6).

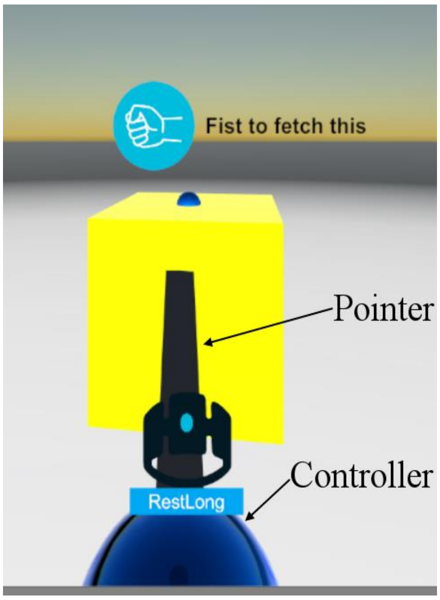

Figure 6: Controller with pointer focusing a virtual object

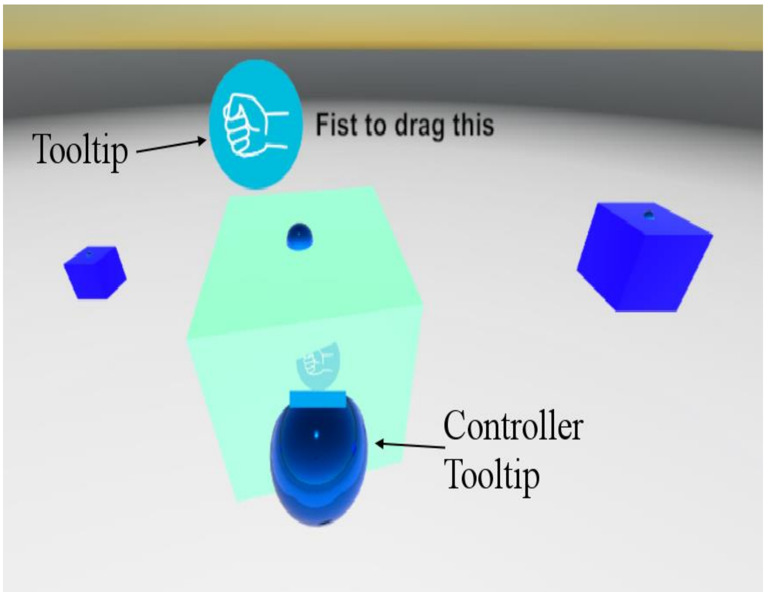

Figure 7: Controller touching a virtual object

The controller is positioned with 70cm distance to the Myo representation along the forward axis. To interact with virtual objects in close range, the user must either touch or focus these virtual objects. A virtual object is touched when the controller is colliding with it (see Figure 7). To focus a virtual object, the user must point at it being in a range of five meters away (see Figure 6). Additionally, the user can only interact with an object if it is interactive and currently not used in another way. While interacting with an object, the user can use different actions (see Table 2) by performing hand gestures.

The central component of the user interaction with the environment is the controller. The main purpose of the controller is to represent the user's hand by extending the user's body with a virtual body, to interact with virtual objects and to display the current state of interaction to the user. In contrast to the Myo representation, the user has a sense of embodiment with the controller. Comparing the results of [12][pp.8–9] an abstract representation for a controller is more effective, but the level of immersion is low and thereby less effective for difficult tasks. The reasons for that is that an abstract representation is providing less feedback in comparison to a realistic representation e.g. a hand model. However, drift, calibration errors and the mismatch of hand rotation and rotation of the arm, are very likely to cause irritation. To reduce irritation and discomfort for the user, an abstract representation was chosen. Therefore, the controller representation is a semi-transparent blue sphere. This abstract representation can neglect the disadvantages of the realistic approach and is easy to use and self-explanatory. Also, the user can see what is behind the controller, for more precise interaction. The size of the sphere is 15cm in diameter (see Figure 7). The Myo is also represented by a colored sphere, with a diameter of 15cm and no transparency.

5.5 Gesture Handling and Mapping

From the five available hand gestures of the Myo SDK (see Figure 2), all five gestures are actively used for the user interface. The five hand gestures are mapped to the previously defined and described actions (see Table 2) as follows:

- Fist is used for the select action.
- Fingers Spread is used for “discard current action” action.
- Wave Left is used for the back action.
- Wave Right is used for the forward action.
- Double Tap is used for the menu action.

The mapping of the actions is based on the results of the Myo evaluation and the suggestions of the Myo Developer guidelines [51]. The gesture mapping minimizes negative effects of unintended actions, by assigning gestures to opposing actions that are unlikely to be misclassified. Moreover, the actions are assigned to gestures most intuitively representing their intentions [51]. To trigger the fast back and forward actions, the user holds the Wave Left or Wave Right gestures for more than 1 second. Also, holding the rest gesture for more than 800 milliseconds, triggers the discard action. The double mapping of gestures for the action is proposed to offer the user more freedom in his or her input selection.

Using the Fist and Fingers Spread gestures in the same context, for selecting and discarding actions, might cause same issues. The evaluation of the Myo showed that the Fist gesture is likely to be misclassified as Fingers Spread (see chapter 5.1). Unfortunately, mapping the Double Tap or one of the wave gestures to discard actions is unintuitive [51] and the Myo does not recognize more gestures [35]. Therefore, the Fingers Spread gesture should be used to discard actions. The user interface was designed to avoid any issues originating from this unfavorable gesture mapping. Every time the user is required to do a selective command, e.g. to grab an object, the current state of interaction cannot be aborted with the discard action. Instead to cancel the interaction the user is supposed to move the controller away from the virtual object.

5.6 Feedback

To keep the feedback clear and most helpful for the user, it is separated by the type of information:

- sounds for user input
- haptic feedback for object contact
- a color scheme for Myo armband states
- text and icons for tooltips describing object interaction and gestural input

When interacting with objects out of sight, haptic and audio feedback for the ongoing interaction is critical for the user. In these situations, the user has no senses available to observe the current interaction. When the controller starts touching an object, the armband vibrates shortly. This feedback indicates whether the user is touching an object and can thereby interact with objects out of sight. For example, the user can grab objects behind his or her back without necessarily looking at the objects, allowing more natural interaction with virtual objects based on haptic feedback. Furthermore, the device is vibrating as well when the user is focusing an interactive object with the pointer. This pattern is easy to understand and provides guidance for object localization and self-localization.

The armband vibrates in one more case if the Myo input is activated or deactivated. Using the Myo’s vibration as another kind of feedback can be confusing for the user, but for this certain purpose the feedback is very important. Also, the user should be able to distinguish the feedback, because changing the Myo state does not happen accidently.

The amount of possible interactions is limited and comprehensible, but further guidance might help the user to explore the capabilities more quickly. Every interactive object has a tooltip, describing the current possible actions and their mapped gesture. If the object is currently not interactive for some reason, the user is informed as well. The tooltips consist of a text field describing the current interaction and an icon for the required gesture. The tooltips are attached to the virtual objects, are positioned above the objects and auto rotate towards the user (see Figure 6).

A color scheme is used to display the current state of the Myo armband to the user. If the device is connected and running, the connector’s shape is colored blue. If the device is not connected or has a critical failure, the connector’s shape is colored red. If the Myo input is disabled, it is colored grey. Due to the connector’s position at the user’s body center, the user only sees the connector rarely during his or her routine. But information regarding armband state is only rarely needed by the user and is therefore positioned out of the user’s area of attention.

Informing the user about the input made and recognized is difficult for multiple reasons. The information must be clear and understandable, accessible in every situation and guide the user in his or her actions. Sound is a great tool to help with this, because sounds can be used to express different states or purposes by exploring common patterns, e.g. for correct input. Furthermore, it is possible to deploy new sounds that are easy to adapt and self-explanatory. For every successful interaction, a distinct sound expressing success, as known from other interfaces, is used and a unique sound for failing interactions is used.

The sounds do not give the user any information about the currently detected gestures; hence the currently detected gesture is displayed with a text and the associated icon at the controller (see Figure 7). Thus, the user can quickly detect misclassified gestures and react, e.g. by performing the intended gesture again or performing another gesture. Additionally, the user can learn how to perform the gestures more effectively by considering the received feedback.

5.7 Further Findings

In the first steps of the UI design, the user was supposed to wear two Myos, one at each forearm. Both Myos were positioned relative to the camera with a fixed offset. Each of the two Myos had a pointer aligned with the forward vector. The user could move the pointer by rotating his or her arms. If the two pointers crossed each other, the controller was positioned at this position and the user could interact with virtual objects. Testing this solution during development showed that users found it very unintuitive and difficult to use. Performing precise tasks was almost impossible and required high concentration.

Furthermore, performing gestural input proved to be difficult as well. Combining the input from both Myos, by performing the same gesture with both Myos within a short period, reduced false-positives but was very time consuming, tiring for the users, and unintuitive. Decoupling the Myos from each other and accepting gestural input from both devices produced too much unintended behavior. Using only a single armband for gesture detection, was the most effective solution but had a few downsides as well:

- Holding both controllers steadily and performing a gesture at the same time required lots of concentration from the user
- Not using the second arm for input, but only to position the controller, made this arm quite useless

In the end, the approach of combining two Myo armbands for the user interface was dropped during development for three main reasons:

- Combination of the Myo armbands gesture detection proved to be ineffective
- Little precision and control in controller positioning by crossing the two Myo pointer
- Controller handling was unintuitive for the user

Enabling the user to throw objects was one of the requirements (see 5.3 Requirements). Unfortunately, this proved to be rather difficult, because releasing the grab mechanic in the correct moment was almost impossible. Furthermore, it was difficult to distinguish if the user wanted to throw an object or not. Often the objects flew around, without the user’s intention.

6 Implementation

A quick overview of the implementation of the proposed user interface, see chapter 5, is given in this chapter.

6.1 Overview

The user interface was designed using C# and the Unity3D game engine with version 2017.02.0f3. Unity3D was chosen for multiple reasons: availability of the Myo SDK for Untiy3D [31], support of the Microsoft HoloLens [52], and multiple other VR setups [53], helpful elements like an editor, a UI system and portability to many platforms and operating systems and a physics engine. The physics engine allows physics based interaction with virtual objects, e.g. by throwing and dropping them. These components offer great comfort and effectiveness in implementation.

C# is used as a programming language because the MyoSDK plugin high API for Unity3D is written in C# and C# is with JavaScript one of the languages natively supported by the Unity3D editor and engine.

The user interface implementation is divided into multiple modules. All the modules are strictly separated by their functions to keep the code basis manageable and clear. Also, the partition guarantees that changes are easily deployed. For the controller and interactive object modules all functionalities, like pointer, touch recognition, tooltips etc., are encapsulated in components. These components can be quickly added to the objects. The components implement interfaces, depending on their purpose. Therefore, it is not necessary to change the implementation of a certain object, because the components are managed automatically.

Communication between the modules is limited to defined access and exit points and reduced to a necessary minimum. The communication between controller and interactive objects is based on events. The event informs the two partners, controller and object, of an interaction, about state changes, errors, etc. (see Figure 8).

```
/// <summary>
/// The object invoking this event. Not nullable.
/// </summary>
public Object sender;

/// <summary>
/// The involved controller for this event, might be null
/// </summary>
public IController controller;

/// <summary>
/// The type of event, e.g. StateChanged.
/// </summary>
public ObjectMessageType eventType;

/// <summary>
/// Additional message, e.g. for debugging.
/// </summary>
public string message;
```

Figure 8: Interactive object event message object

6.2 Structure

The input from the device is gathered by the MyoSDK. Tracking of the accelerometer and absolute device orientation are directly accessible. Hand gestures are classified on the device and can be pulled from the SDK as well. Additionally, the SDK handles communication with the Myo via Bluetooth, e.g. for vibration commands or information on the device state.

The user input is interpreted in the Myo handling module. The device orientation is recalculated and transformed into to quaternions. Based on the device's orientation the position of the controller is calculated and updated every frame. The recalibration process is implemented in the Myo handling module. A main work of the Myo handling module is to detect additional gestures, like the hold gestures, map all gestures to actions and report them to the controller (see Figure 9). By first mapping the gestures and then reporting the actions to the controller, it is simple to change the gesture mapping and use the rest of the implementation with other devices.

The controller is the abstract representation of the virtual interaction tool. It handles the current state of interaction and displaying this state to the user. All user actions are handled by the controller (see Figure 9). The actions are used on currently selected objects, by passing the action to the object. The controller recognizes interactive objects by Rigidbody collision detection and checking if the pointer is focusing an interactive object. The tooltip is handled by the controller (see Figure 7). All components of the controller can be easily changed or deactivated without affecting the other components.

Every interactive object handles the effect of a user action by its own. Furthermore, it explains the effect of certain action to use with another tooltip (see Figure 6). The effects can be changed easily by adding or removing components to the object and adjusting the command handling. Not all actions are necessarily applicable for an object. Therefore, the result of an interaction is reported to the calling controller. For example, an object can only be used by one controller at a time.

```
public interface IController {
    /// <summary>
    /// The active gesture Mapping for this controller.
    /// </summary>
    Mapping ActiveMapping { get; }

    /// <summary>
    /// Transform handler for this controller. Handling position and rotation of the controller.
    /// </summary>
    ControllerTransform Transform { get; }

    void Initialize(Mapping activeMapping, InputHandling.InputHandler inputHandler);

    /// <summary>
    /// The input handler detected a valid user input.
    /// </summary>
    /// <param name="command">The abstract user action. </param>
    bool HandleCommand(Mappings.Command command);

    void Destroy();
    void Hide();
    void Show();
}
```

Figure 9: Controller interface, as accessible to all other modules of the user interface

7 User Study

To evaluate the user interface a user study was conducted. The proceedings, results, and discussion of the study are presented in this chapter.

7.1 Goal

The goal of the user study is to evaluate the usability of the designed user interface and the ability to accomplish the defined user stories (see Table 1). Usability is defined as the user's ability to achieve certain goals in a certain context with a user interface [54]. To measure the usability of the user interface, the three main components of usability, user acceptance, efficiency, and effectiveness, were evaluated (see Figure 10) [55]. Furthermore, the user study helps to identify issues, strengths and weaknesses of the user interface and areas of high interest to focus on in future works.

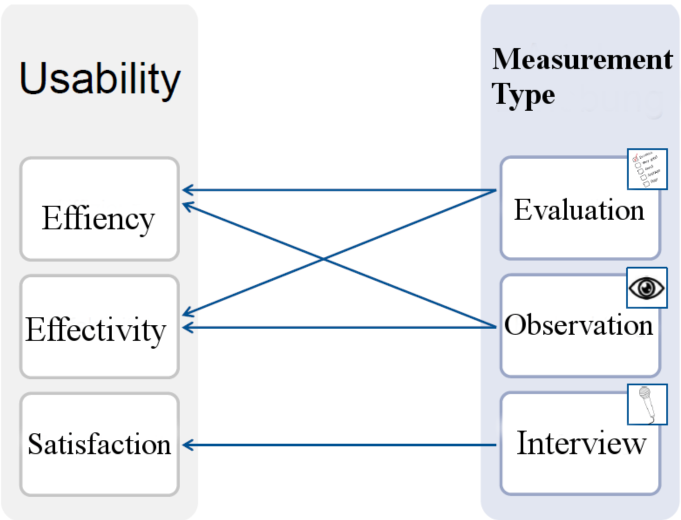

Figure 10: Usability measurement explained, translated from [56]

7.2 Methodology:

The user study was conducted in the augmented reality lab of the chair for Computer Aided Medical Procedures & Augmented Reality at the Department for Computer Science at the Technical University of Munich. The test group were students, mainly with a background in computer science. Most of the students are ranging from 18 - 29 years of age and mixed sexes. A total of 18 persons participated in the user study.

The experiment was set in a virtual reality environment with a multitude of tasks for the test persons to solve. These tasks were based on the user stories, e.g. touching objects in multiple distances or grabbing close objects. The procedure of the experiment was the same for every test person. Before the tasks began, every tester was supposed to learn the hand gestures used by the interface, as the testers did for the Myo evaluation.

In total solving all the tasks took about 15 to 20 minutes for each participant and gesture training and setting up the experiment for a person additional 10 minutes.

To evaluate the effectiveness and efficiency of the proposed user interface, the time taken to solve a task and failures or inability to solve the given tasks were measured. Additionally, during the entire experiment the gestures, timings etc. were logged as well. The format of the logs was as described for the Myo evaluation. After every task, the user was asked about the perceived difficulty level of the task. Also, the user could report issues occurring while solving the task, e.g. difficulties to point at an object and use a wave gesture simultaneously.

Mainly to explore the user acceptance of the new interface, but also to get data on efficiency and effectiveness of the user interface, interviews with the persons taking part in the user study were conducted after the experiment. The interview was used to receive the System Usability Scale (SUS), of the user interface (see Appendix 1). The System Usability scale is a simple way to assess the global usability of a system. It is expressed in a score ranging from 0 to 100, indicating the quality of the evaluated user interface [55]. The interview took about 10 minutes and was unsupervised.

Before performing a task, the testers realigned the orientation of the controller, to eliminate any kind of drift from the experiment. To reduce the impact of virtual reality environment on the experiment, all tasks were designed to use no kind of locomotion, except the free movement of the person. This movement was further reduced in task design.

As previously stated, the users learned the gestures but no functions of the user interface were explained or mentioned to the users. Only the realignment procedure was explained. Before the tasks started, every test user explored the user interface within a warm-up area. They were proposed to try the gestures on objects and to observe the behavior.

From every test user age and sex was collected. For the detailed experiment procedure and description of the task see

Appendix 2.

7.3 Limits of Methodology

Extending the results of the user study on a greater range of persons is not possible, because the test users were a very homogenous group mainly consisting of 18 – 29 years old males, studying computer science. By realigning the controller orientation before every single experiment, drift was eliminated from the user study, thus, it is not possible to make any remarks regarding rotational drift for the Myo armband.

It is also important to consider that the test users were not familiar with object interaction in virtual environments and. Therefore. Could not compare the proposed user interface with other ones. The limited time for testing made it also difficult to learn the gestures for the Myo accurately and explore the user interface in depth.

Any considerations towards the sense or virtual embodiment by the users are limited, because no data targeting virtual embodiment was collected.

7.4 Results

For all results no significant deviation regarding sex or age was found. None of the test users was familiar with the Myo. The SUS total score has a median of 78.5 and upper quantile is at 85 and the lower one at 72.4 (see Figure 12). The total average score of the SUS is 81. The total score for the system usability scale was calculated with the Equation 1.

SUS = $(\sum_{i=1}^{10} x_i - 1) * 2.5$

Equation 1: Calculation of the SUS total Score, based on classification for an SUS of Bangor [57]

Furthermore, the relatively low σ shows an overall confidence of the test users in the good rating of the interface. Only the values for question 5, “I think, the various functions of the interface are easily explored.”, deviates more strongly. A detailed analysis shows a portion of roughly 30% did not agree on this question. The values for questions 1 and 7 are lower 0.35 than the average for all questions of 4.23 (see Figure 11).

The results show a correlation between the time taken by a test user to solve a task and the perceived difficulty level of this task (see Figure 13 & Figure 14). More difficult tasks took more time to be accomplished (see Figure 13 & Figure 14). The standard deviation, σ, for these tasks with regard on time taken and difficulty level is high. Also, a strong correlation between not reporting issues for a task and low difficulty levels exists. The data further shows that test users reporting issues with a task did not necessarily perceive the task to be difficult. It is important to further analyze the reported issues here (see Figure 13 & Figure 14).

For the tasks 2,4,6,8, and 9 (see Figure 13) an in-depth analysis, considering the before mentioned correlations, was done. These tasks perceived difficulty ranges from medium difficulty 2.5 to 3.5 (see Figure 13). The tasks 2 and 6 are heavily associated with rotational actions and the tasks 4 and 8 are tasks where the users were not able to look while accomplishing the tasks. The tasks 4 and 8 were perceived significantly more difficult than their non-handicapped correspondents, with the difficulty level rising by 2.5 (see Figure 13) and almost every tester reported issues. The main issues reported were:

- audio feedback confusing, or not understood at all
- haptic feedback too inaccurate, the test users were not aware if the pointer or controller was on the object or close by
- unclear object state regarding whether the object currently was being grabbed or dropped
- unclear if the fingers spread gestures was detected by the Myo

For the tasks 2 and 6, the test users reported often difficulties with the wave gestures. Pointing at an object and performing the wave gestures was perceived difficult. Moreover, holding an object and rotating it at the same time was also tough for the test users.

The last task, number 9, was perceived with a medium difficulty of 2.4 and a good average time of 42.2 seconds to solve the task. But 12 of the 18 test users reported issues, solving the exercise of building a tower. Most of the reported issues were regarding precision, the test users reported lack of capabilities to place objects accurately, to rotate them precisely and awkward object rotation.

Almost everyone was able solve all nine tasks within time, only 4 events occurred when a test user was not able to solve a certain task. This happened twice for the tasks 4 and 8.

The tasks 1, 3, 5, 7 were perceived to be very easy and quickly accomplished by the test users. Also, the results only slightly deviate for these tasks (see Figure 13 & Figure 14).

During the user study, misclassification of gestures happened frequently, often the Fist and Fingers Spread gesture were misinterpreted by the Myo.

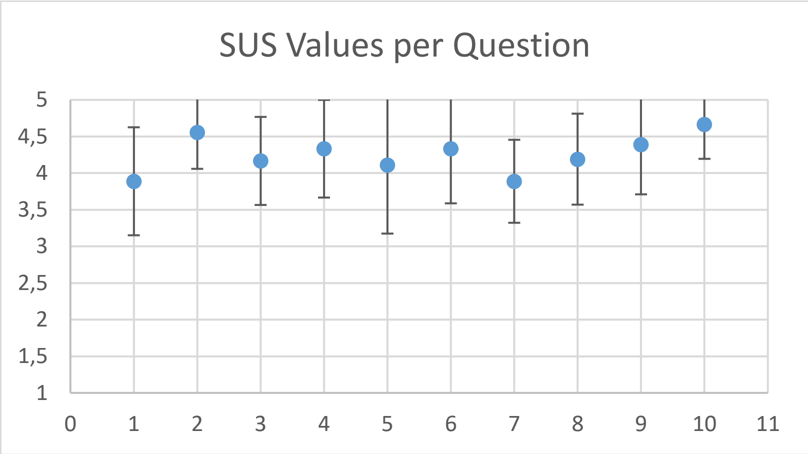

Figure 11: System usability scale values for each question normalized to a range of 1 to 5 with standard deviation, higher is better

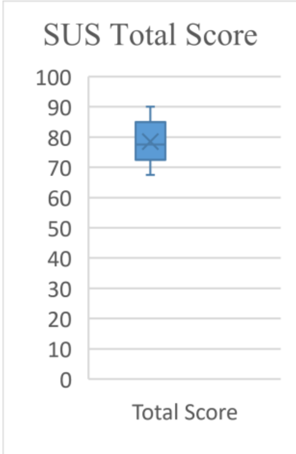

Figure 12: SUS Total Score, higher is better

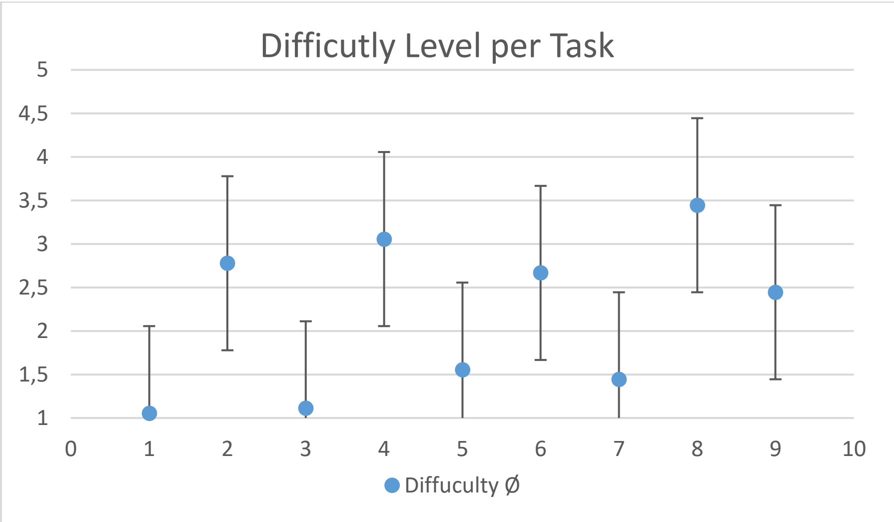

Figure 13: Test users perceived difficulty level per task, lower is better

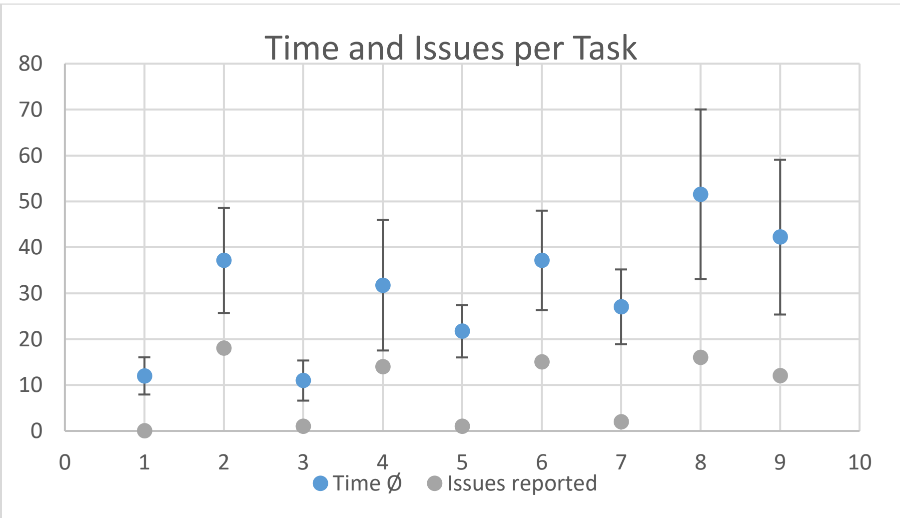

Figure 14: Test users time in seconds to solve a task and the number of users reporting issues for a specific task, lower is better

7.5 Discussion of Results

First, it is important to point out that the user interface could accomplish all the previously defined user stories (see Table 1). The test users could solve the given tasks based on the user stories. The user interface can be regarded as intuitive, because the users fulfilled the tasks without explanations and explored the user interface on their own. The claim is further supported by the good value of 4.1 for item 5.

The pointer, touch and grab mechanics work properly as the results show. The users could quickly hit even far distant and small objects and grab these. Combing the pointer with the rotation proved to be rather complicated. Many users were not able to, or had severe problems performing the wave gestures and, therefore, had difficulties in rotating and precisely handling objects. Also, dropping grabbed objects was more of an issue, because the fingers spread gesture was often not detected and resting the hand for a certain period was often explored quite lately. Offering a second option to discard the current action was very helpful for the users because they could choose the most effective one for themselves.

The great performance with the pointer and the ability to localize objects and interact with them without sight indicates the users were incorporating the controller into their body schema. Furthermore, these results indicate the users virtually embodied the controller.

The user study showed multiple issues with the user interface, especially, the feedback was not providing the needed information. Because the audio feedback was often not understood or even properly recognized by the users, the difficulty of the non-view tasks was much higher than expected. A simple feedback for succeeding and failing actions is not enough for multiple reasons:

The users were not aware of the actual state of the object, was the object dropped, rotated etc., if they cannot see it. The users often had to guess whether they grabbed an object or not.

The sounds were not naturally associated with the actions or situations; hence the user did not understand the relationship between the audio and the current actions.

The haptic feedback was also not working as intended, because it was too inaccurate and not diverse. Vibrating when the user focused an object with the pointer or touched it was not always helpful, because for the user it was difficult to differentiate whether he or she was touching or focusing an object or already passed it. Due to the issues with the feedback mechanics, the performance for tasks 4 and 8 significantly decreased. Moreover, the unclear object state further decreased the efficiency of the user interface.

A total SUS score of nearly 81 is surprisingly high. This might be due to a high sense of embodiment with the controller, resulting in a high emotional attachment with the user interface and, therefore, better ratings by the test users. The user study, especially task 9, proved the interface to be lacking efficiency. A multitude of users found the interface to be clumsy to a medium extent (see Figure 11 item 7). Also, the lack of free movement of the controller reduced the precision significantly. The users had difficulties precisely rotating objects and positing them accurately, because they were not able to rotate the controller independently from their arm. It was often necessary to drop the object, rotate it with the wave gestures and grab it again.

The user study shows the proposed user interface is effective and users are satisfied as well, but efficiency is lacking. This lack of efficiency needs to be tackled in future works as well as the insufficient feedback mechanics.

8 Conclusion

Developing a user interface for virtual object interaction in virtual or augmented realities with the Myo is possible and the results are promising. The results of the user study show that the proposed user interface is satisfying for the users. The proposed user interface is easily explored. The users could learn the functionalities within a few minutes and utilize them for object manipulation. Moreover, even complex tasks, like building a tower of multiple cubes or moving objects around without looking at them at all, could be done.

The evaluation of the Myo provided crucial information to design the user interface according to the explored strengths and weaknesses of the Myo. Using the accurate rotation tracking of the armband to implement a pointer as a fundamental part of the UI worked out very well. The user study showed great performance ratings for pointer interactions with virtual objects. Only the use of the wave gestures in combination with the pointer did not work out.

Developing a failure resistant user interface was particularly important, because as the evaluation showed, the user’s intention is often falsely interpreted. Gestures were only correctly detected with a rate of 66%, which is extremely low, compared to other classification algorithms. Therefore, unintended actions are happening frequently and are disturbing the workflow. Reducing the amount of gestures needed for an action and for opposing actions to avoid mapping gestures that are most frequently being mistaken was a successful solution.

Furthermore, the results for the gesture classification accuracy can serve as a benchmark for other gesture classification approaches in future using the Myo sensors. In my opinion, new gesture classification algorithms are desperately needed for the Myo. Otherwise it is not possible to use the device in efficient user interfaces, compared with controllers, like the Vive or Oculus controller.

A main issue for lacking precision and, in general, for low efficiency in object interaction was the missing capability to freely move the arm and hand. Also, the lacking hand rotation is likely to reduce the sense of control over the controller. As shown in the evaluation, it is not possible to use the accelerometer of the Myo armband for any kind of absolute spatial tracking. However, using the Myo with the HTC Vive offers the possibility to use additional tracker attached to the armband for spatial tracking. The user interface would lose mobility and compatibility but heavily increase user control and accuracy. I suppose that this gain in accuracy results in efficiency and higher sense of body control, and ownership for the user. Another approach would be to use inverse-kinematics [58] to calculate the arm and hand position, by using two Myo armbands on one arm.

The sense of embodiment and its effect on the user interface were not measured in the user study. Nonetheless, actively considering virtual embodiment in the user interface design had an important impact on the concept of pointer design, mapping of gestures, and overall control of the user interface. Without the utilization of virtual embodiment, the pointer most likely would not have worked, because it requires a high degree of body localization to be used efficiently. Gesture control and orientation tracking can be used for virtually embodied user interfaces, therefore, the Myo armband is applicable for these tasks.

Haptic feedback is a strong tool that needs to be used more intensely with the Myo armband. Currently, two main issues remain, developing feedback patterns for the user and applying these custom patterns to the device. Many learnings from other interface, especially for video games, e.g. with rumble Controller or the Nintendo Switch, can be included for developing more advanced tactical feedback, especially feedback for correct object localization. To allow natural ways of object interaction, it is necessary to improve the feedback on the state of interaction, mainly for grabbing and dropping objects. These actions are likely to be done without continuous oversight through the user.

Implementing the extended Bluetooth API Thalmic Labs provided is worth effort in my opinion, because better feedback will grant the user more awareness of the current state of interaction and object localization. This information is crucial to improve the efficiency of the interface.

The user interface might be deployed in more complex environments like the “robo-lab” to allow the users to interact with these environments in an effective and satisfactory way.

9 Bibliography

[1] HTC, "Vive," 2017, [Online]. Available: https://www.vive.com/de/. [Accessed 19 November 2017].

[2] M. Rani, "These are the companies investing most aggressively in AR and VR.," 28 April 2017. [Online]. Available: https://www.recode.net/2017/4/28/15376268/facebook-augmented-virtual-reality-linkedin-jobs-charts. [Accessed 21 September 2017].

[3] I. Wright, "What Can Augmented Reality Do for Manufacturing?" What Can Augmented Reality Do for Manufacturing?," 11 May 2017. [Online]. Available: https://www.engineering.com/AdvancedManufacturing/ArticleID/14904/What-Can-Augmented-Reality-Do-for-Manufacturing.aspx. [Accessed 21 September 2017].

[4] Microsoft, "Why Hololens," 2017. [Online]. Available: https://www.microsoft.com/de-de/hololens/why-hololens. [Accessed 11 November 2017].

[5] Thalmic Labs, "Tech Specs," 2016. [Online]. Available: https://www.myo.com/techspecs. [Accessed 06 November 2017].

[6] W. Steptoe, A. Steed and M. Slater, "Human tails: ownership and control of extended humanoid avatars.," *IEEE transactions on visualization and computer graphics, 19(4),* pp. 583-590, 2013.

[7] A. Maravita and A. Iriki, "Tools for the body (schema)," *In Trends in Cognitive Sciences, Volume 8, Issue 2,* pp. 79-86, February 2004.

[8] M. Slater, V. Linakis, M. Usoh, R. Kooper and G. Street, "Immersion, Presence, and Performance in Virtual Environments: An Experiment with Tri-Dimensional Chess," *ACM virtual reality software and technology (VRST), Vol. 163,* pp. 163-172, July 1996.

[9] Virtual Bodyworks, "Virtual Reality: Virtual Embodiment," 8 November 2016. [Online]. Available: https://www.digitalcatapultcentre.org.uk/virtual-reality-virtual-embodiment/. [Accessed 21 September 2017].

[10] K. Kilteni, J. M. Normand, M. V. Sanchez-Vives and M. Slater, "Extending body space in immersive virtual reality: a very long arm illusion," *PloS one, 7(7), e40867,* pp. 1-15, July 2012.

[11] K. Kilteni, R. Groten and M. Slater, "The sense of embodiment in virtual reality.," *Teleoperators and Virtual Environments, Jg., Nr. 4,* pp. 373-387, 2012.

[12] F. Argelaguet, L. Hoyet, M. Trico and A. Lecuyer, "The role of interaction in virtual embodiment: Effects of the virtual hand representation.," in *2016 IEEE Virtual Reality (VR). doi:10.1109/vr.2016.7504682*, 2016.

[13] G. Kamen, "Electromyographic Kinesiology," in *Research Methods in Biomechanics*, Champaign, Human Kinetics, 2004.

[14] E. A. Clancy and N. Hogan, "Multiple site electromyograph amplitude estimation," *IEEE Transactions on Biomedical Engineering, vol. 42, no. 2,* pp. 203-211, February 1995.

[15] R. Merletti, D. Farina and M. Gazzoni, "The linear electrode array: A useful tool with many applications," *Journal of electromyography and kinesiology: official journal of the InternationalSociety of Electrophysiological Kinesiology., vol. 13, no. 1,* pp. pp. 37-47, December 2002.

[16] J. Winkel and K. Jørgensen, "Significance of skin temperature changes in surface electromyography," *European journal of applied physiology and occupational physiology 63.5,* p. 345–348, 1991.

[17] T. A. Kuiken, M. M. Lowery and N. S. Stoykov, "The effect of subcutaneous fat on myoelectric signal amplitude and cross-talk," *Prosthetics and Orthotics International, vol. 27,* pp. 48-54, August 2003.

[18] Boxtel, "Optimal signal bandwidth for the recording of surface EMG activity of facial jaw oral and neck muscles," *Psychophysiology, vol. 38, no.1,* pp. 22-34, January 2001.

[19] I. Mendez, B. W. Hansen, C. M. Grabow, E. J. Smedegaard, N. B. Skoggberg, X. J. Uth and E. N. Kamavuako, "Evaluation of the Myo armband for the classification of hand motions," *International Conference on Rehabilitation Robotics (ICORR), doi:10.1109/icorr.2017.8009414,* pp. 1211-1214, 2017.

[20] H. Huang, P. Zhou, G. Li and T. Kuiken, "Spatial Filtering Improves EMG Classification Accuracy Following Targeted Muscle Reinnervation," *Annals of Biomedical Engineering, Volume 37, Issue 9,* p. 1849–1857, September 2009.

[21] C. Castellini and P. van der Smag, "Surface EMG in advanced hand prosthetics," *Biological cybernetics, 100(1),* pp. 35-47, 2009.

[22] A. Alkan and M. Günay, "Identification of EMG signals using discriminant analysis and SVM classifier.," *Expert Systems with Applications, vol. 39, no. 1, doi:10.1016/j.eswa.2011.06.043,* p. 44–47, 2012.

[23] M. F. Lucas, A. Gaufriau, S. Pascual, C. Doncarli and D. Farina, "Multi-Channel surface EMG classification using support vector machines and signal-Based wavelet optimization.," *Biomedical Signal Processing and Control, vol. 3, no. 2, doi:10.1016/j.bspc.2007.09.002,* p. 169–174, 2008.

[24] A. Rahimi, S. Benatti, P. Kanerva, L. Benini and J. Rabaey, "Hyperdimensional biosignal processing: A case study for EMG-Based hand gesture recognition.," in *2016 IEEE International Conference on Rebooting Computing (ICRC)*, San Diego, CA, 2016.

[25] A. Rahimi, P. Kanerva and J. M. Rabaey, "A robust and energy efficient classifier using brain-inspired hyperdimensional computing,," *Proceedings of the 2016 International Symposium on Low Power Electronics and Design.,* pp. 64-69, August 2016.

[26] M. B. I. Reaz, M. S. Hussain and F. Mohd-Yasin, "Techniques of EMG signal analysis: detection, processing, classification and applications," *Biological Procedures Online, vol 8, iss 1,* pp. 11-35, 01 December 2006.

[27] CBC News, "Myo gesture-control band for sale on Amazon," 04 March 2015. [Online]. Available: http://www.cbc.ca/news/canada/kitchener-waterloo/myo-gesture-control-band-for-sale-on-amazon-1.2979606. [Accessed 11 November 2017].

[28] P. Bernhardt, "Myo | #MyoCraft: Logging IMU and Raw EMG Data," 14 March 2015. [Online]. Available: http://developerblog.myo.com/myocraft-logging-imu-and-raw-emg-data/. [Accessed 11 November 2017].

[29] P. Bernhardt, "Myo | #MyoCraft: Using Multiple Myo Armbands," 16 May 2015. [Online]. Available: http://developerblog.myo.com/myocraft-using-multiple-myo-armbands/. [Accessed 06 November 2017].

[30] Thalmic Labs, "Myo Developer FAQ," 8 October 2015. [Online]. Available: https://developer.thalmic.com/forums/topic/255/. [Accessed 06 November 2017].

[31] P. Bernhardt, "Myo | Setting up the Myo Package in Unity," 19 December 2014. [Online]. Available: http://developerblog.myo.com/setting-myo-package-unity/. [Accessed 06 November 2017].

[32] Thalmic Labs, "Getting Started," 2014. [Online]. Available: https://developer.thalmic.com/docs/api_reference/platform/getting-started.html. [Accessed 05 January 2017].

[33] S. Rawat, S. Vats and P. Kumar, "Evaluating and exploring the MYO ARMBAND," in *2016 International Conference System Modeling & Advancement in Research Trends (SMART). doi:10.1109/sysmart.2016.7894501*, 2016.

[34] P. Bernhardt, "Myo Scripting Basics Part Five: Shake It," 14 October 2014. [Online]. Available: http://developerblog.myo.com/getting-started-myo-scripts-part-5/. [Accessed 21 September 2017].

[35] Thalmic Labs, "What gestures does the Myo armband recognize?," [Online]. Available: https://support.getmyo.com/hc/en-us/articles/202647853-What-gestures-does-the-Myo-armband-recognize-. [Accessed 18 September 2017].

[36] M. Abduo and M. Galster, Myo Gesture Control Armband for Medical Applications, 2015.

[37] Thalmic Labs, "Myo Bluetooth Protocol," 28 September 2015. [Online]. Available: https://github.com/thalmiclabs/myo-bluetooth. [Accessed 19 December 2017].

[38] H. SHAABAN, C. PEREIRA, R. WILLIAMS and V. C. LEES, "THE EFFECT OF ELBOW POSITION ON THE RANGE OF SUPINATION AND PRONATION OF THE FOREARM," *Journal of Hand Surgery (European Volume),* pp. Vol 33, Issue 1, pp. 3 - 8, 01 February 2008.

[39] U.S. Department of Health & Human Services,, "User Interface Design Basics," [Online]. Available: https://www.usability.gov/what-and-why/user-interface-design.html. [Accessed 11 November 2017].

[40] Treehouse, "10 User Interface Design Fundamentals," 7 August 2012. [Online]. Available: http://blog.teamtreehouse.com/10-user-interface-design-fundamentals. [Accessed 06 November 2017].

[41] J. J. Garret, Elements of user experience, the: user-centered design for the web and beyond., Berkeley, CA 94710: Pearson Education, 2011.

[42] M. B. Rosson and J. M. Carroll, Usability engineering: scenario-based development of human-computer interaction, San Francisco, CA: Morgan Kaufmann, 2001.

[43] R. Hartson and P. Pyla, "Design Thinking Ideation and Sketching," in *The UX Book: Process and Guidelines for Ensuring a Quality User Experience*, Morgan Kaufmann (Elsevir), 2012, pp. 251-298.

[44] M. Richter and M. D. Flückiger, Usability Engineering kompakt: benutzbare Produkte gezielt entwickeln, p. 40 - 67, Springer-Verlag, 2013.

[45] T. Brinton, "User Stories" A Foundation for UI Design," 17 March 2015. [Online]. Available: http://www.uxbooth.com/articles/user-stories-a-foundation-for-ui-design/. [Accessed 06 November 2017].

[46] O. Blanke, M. Slater and A. Serino, "Behavioral, neural, and computational principles of bodily self-consciousness," *Neuron, vol. 88,* pp. 145-166, 29 September 2015.

[47] B. Lenggenhager, T. Tadi, T. Metzinger and O. Blanke, "Video ergo sum: manipulating bodily self-consciousness.," *Sciene, vol. 317,* pp. 1096-1099., 24 August 2007.

[48] M. Botvinick and J. Cohen, "Rubber hands 'feel' touch that eyes see," *Nature 391, 756,* p. 756, 19 February 1998.

[49] F. Vignemont, "Habeas Corpus: The Sense of Ownership of One's Own Body," *Mind & Language, vol. 22, iss. 4,* pp. 427-449, September 2007.

[50] F. Vignemont, "Bodily Awareness," Stanford University, 07 October 2015. [Online]. Available: https://plato.stanford.edu/entries/bodily-awareness/#SenBodOwn. [Accessed 09 January 2018].

[51] Thalmic Labs, "Myo Developer User Experience Guidelines," 27 November 2014. [Online]. Available: https://developer.thalmic.com/ux/. [Accessed 16 September 2017].

[52] Microsoft, "Unity development overview," Microsoft, 2017. [Online]. Available: https://developer.microsoft.com/en-us/windows/mixed-reality/unity_development_overview. [Accessed 11 November 2017].

[53] Unity Technologies, "VR overview," Unity Technologies, 8 December 2017. [Online]. Available: https://docs.unity3d.com/Manual/VROverview.html. [Accessed 11 November 2017].

[54] International Organization for Standardization, *Ergonomic Requirements for Office Work with Visual Display Terminals, DIN EN ISO 9241-11:1998, 3.1,* Geneva, 1998.

[55] J. Brooke, "SUS-A quick and dirty usability scale.," *Usability evaluation in industry, 189. Jg., Nr. 194,* pp. 4-7, 1994.

[56] K. Bengler and M. Wagner, *Usability, Usermodellierung und Software-Ergonomie,* Garching: Technische Universität München, 2015.

[57] A. Bangor, J. Miller and P. Kortum, "Determining What Individual SUS Scores Mean: Adding an Adjective Rating Scale," *Journal of Usability Studies, vol.4, Issue 3,,* pp. 114-123, May 2009.

[58] A. Leitner, "Controlling a Physics Based Mech Simulation in Room-Scale Virtual Reality," Department of Informatics Technical University of Munich, Garching, 2017.

10 Appendix

10.1 Appendix 1: System usability scale questionnaire

		Strongly disagree				Strongly agree
1.	I would like to use this interface more frequently for this purpose. [Ich würde dieses System zu diesem Zweck häufiger verwenden.]	1	2	3	4	5
2.	I found the interface unnecessary complex. [Ich fand die Oberfläche unnötig komplex.]	1	2	3	4	5
3.	I thought the interface to be easy to use. [Ich fand das System war leicht zu bedienen.]	1	2	3	4	5
4.	I think, I would need support from an expert to be able to use this interface. [Ich denke, Ich würde die Unterstützung durch einen Experten benötigen um diese Oberfläche zu benutzen.]	1	2	3	4	5
5.	I think, the various functions of the interface are easily explored. [Ich fand die verschiedenen Funktionen der Oberfläche sind einfach zu entdecken.]	1	2	3	4	5
6.	I found the interface to be inconsistent. [Ich fand die Oberfläche inkonsistent.]	1	2	3	4	5
7.	I think many people would easily learn how to use this interface.					

	[Ich glaube, dass die meisten Menschen sehr schnell lernen würden, mit dem System umzugehen.]	1	2	3	4	5
8.	I found the interface to be clumsy. [Ich fand die Oberfläche umständlich zu benutzen.]	1	2	3	4	5
9.	I felt comfortable using this interface. [Ich fühlte mich bei der Nutzung der Oberfläche sicher.]	1	2	3	4	5
10.	I had to learn a lot, before I was able to use this interface. [Ich musste viele Dinge lernen, bevor ich mit dem System arbeiten konnte.]	1	2	3	4	5

10.2 Appendix 2: User study procedures

Prelude:

Actions:

- Explanation of the experiment – 180 seconds
- Sync Myo and warm up Myo – 180 seconds
- Train gestures – 300 seconds

Time:

- Explanation of the experiment – 180 seconds
- Sync Myo and warm up Myo – 180 seconds
- Train gestures – 300 seconds
- **Total: 680 seconds**

Task 00 – Warm up:

Setting:

- Empty room with 4 objects of different sizes and distance to the user
- The objects are of these sizes: 0.5 meters, 0.3 meters, 0.15 meters
- All the objects are cubes

Task:

- The user is supposed to try the gestures, observe the behavior of the interface and in general get familiar with the UI.
- No hints or explanations are given to the user.

Time:

- 180 seconds

Task 01 – Train targeting:

Setting:

- Empty room with 12 objects in different distances to the user
- 1 meter distance
- 2 meter distance
- 3 meter distance
- 5 meter distance
- Every four of the objects the same sizes: 0.5 meters, 0.3 meters, 0.15 meters

- All of the objects are cubes

Task:

- The user is supposed to focus the different objects with the pointer one time without moving more closely

Time:

- 60 seconds

Task 02 – Simple Rotation
Setting:

- Empty room with 6 objects in different distances to the user
- 5 meter
- 3 meter
- All objects are cubes and have the same size of 0.5 meters, 0.3 meters

Task:

- The user is supposed to rotate the left cubes left around by 180 degree and the right ones by 120 degree right

Time:

- 45 seconds

Task 03 – Simple Grab
Setting:

- Empty room with 2 objects very close to the user

Task:

- Grab and drop the objects at the same spot multiple times

Time:

- 30 seconds

Task 04 – No View Grab
Setting:

- Empty room with 1 object with 1 meter distance to the user

Task:

- Grab and drop the objects at the same spot multiple times without viewing the action

Time:

- 30 seconds

Task 05 – Drag and look closely

Setting:

- Empty room with 2 objects with 3 meter distance to the user

Task:

- Fetch and drag to the user to have a closer look

Time:

- 30 seconds

Task 06 – Drag and examine

Setting:

- Empty room with 2 objects with 4 meter distance to the user

Task:

- Fetch and drag to the user to have a closer look by rotating the objects and looking at the bottom of the objects

Time:

- 60 seconds

Task 07 – Drag around and drop

Setting:

- Empty room with 3 objects with 2 meter distance to the user

Task:

- Grab and drag the object to multiple target positions, drop them there

Time:

- 90 seconds

Task 08 – No View Drag around and drop

Setting:

- Empty room with 3 objects with 2 meter distance to the user

Task:

- Grab and drag the objects to a position, drop them there without viewing the action

Time:

- 150 seconds

Task 9 – Build a tower

Setting:

- Empty room with 6 objects with 2-meter distance to the user

- Every two of the objects have the same sizes: 0.5 meters, 0.3 meters, 0.15 meters

Task:

- Build a tower consisting of at least three objects

Time:

- 90 seconds

Afterwards:

Action

- The users will fill out the SUS questionnaire on their own.

Time:

- 300 seconds

10.3 Appendix 3: Scenarios for the User Interface, selected by interest

Scenario Element	Scenario 01	Scenario 02	Scenario 03	Scenario 04	Scenario 05	Scenario 06
Setting	Augmented reality in an office environment	Augmented reality in an office environment with a personal computer	Augmented reality in an office environment with a personal computer	Augmented reality in an office environment with a personal computer	Augmented reality in an office environment with a personal computer	Augmented reality in an office environment with a personal computer
Actors	Researcher with experience in AR and with the Myo armband	Researcher with experience in AR and with the Myo armband	Researcher with experience in AR and with the Myo armband	Researcher with experience in AR and with the Myo armband	Researcher with experience in AR and with the Myo armband	Researcher with experience in AR and with the Myo armband
Task Goal	Examine properties of a virtual robot	Navigate the robot with the computer controls	Reset the toppled robot without moving to the position of the robot, because the place is blocked	Drop an object into the path of a self-moving robot	Grab an object while monitoring the collision avoidance from the robot.	Throw an object into the path of the robot.
Plans	slowly rotate the robot clockwise to examine its outer appearance, lift it up to examine it from the bottom	Place the robot on the floor, disable myo and then navigate with the keyboard keys	Move to the desired position of the robot, fetch it and drop it at the position	Look around for interactive virtual objects close by, grab a virtual object and drop it at a critical point of the robot’s path, indicated to the researcher	Sense a virtual object close by, grab it virtual object, bring it into sight and drop it at another critical point of the robot’s path, indicated to the researcher	Look around for interactive virtual objects close by, grab a virtual object, look for a target area and throw the object in the desired direction
Evaluation	too slow rotation is frustrating and with too fast rotation the examiner might miss details	missing feedback for object interaction, not disabling the Myo leads to unintended behavior	recalibration might be unclear or failing, a highlighted object indicates that it is focused/touched	a highlighted object indicates that it is focused/touched,	a highlighted object indicates that it is focused/touched, finding the object without requiring too much concentration,	a highlighted object indicates that it is focused/touched, releasing the grab in the right moment can be difficult
Actions	Touch the robot and rotate it clockwise, drag the model above the user head	Grabbing the robot, moving the robot near to the floor and release the model	Enable the Myo, recalibrate the Myo, move to the target position, focus the robot with the pointer, fetch it, move the robot	Move to an object, touch it with the controller, grab the object, move the controller while holding the object to intended position and drop it,	Move the controller around by moving the user's arm, until it hits a virtual object, grab the object, move the controller while holding	Move to a object, touch it with the controller, grab the object, create a trajectory with the controller by moving the arm and release

			near to the floor and release the model		the object to intended position and drop it	the grab in the correct moment
Events	haptic or auditory feedback when touching the model, rotating robot, auditory or textual feedback when interacting with the robot model	haptic or auditory feedback, when touching the robot, haptic or auditory feedback when grabbing and dropping the robot, haptic and visual feedback when disabling the myo, haptic feedback from the keyboard	haptic and visual feedback when enabling the myo, haptic or auditory feedback, when focusing the robot, haptic or auditory feedback when fetching and dropping the robot	haptic or auditory feedback, when touching the object, haptic or auditory feedback when grabbing and dropping the object, visual feedback on the controller while grabbing the object	haptic or auditory feedback, when touching the object, haptic or auditory feedback when grabbing and dropping the object, visual feedback on the controller while grabbing the object	haptic or auditory feedback, when touching the object, haptic or auditory feedback when grabbing and releasing the object, visual feedback on the controller while grabbing the object