Project: ROTOR VR

Robothespian Teleoperation in Virtual Reality

Nikolaos Tsikinis
University College London
18112000

This dissertation is submitted as part requirement for the MRes Degree in 'Virtual Reality', at University College London. It is substantially the result of my own work except where explicitly indicated in the text. The report may be freely copied and distributed provided the source is explicitly acknowledged.

Department of Computer Science
University College London

September 6, 2019

Word count: thirteen thousand, four hundred and eighty nine (13489)

Abstract

Tasks performed by remotely operated (teleoperated) robots can be useful in many ways, with many studies over the last years investigating different methods of implementing such systems. In cases where the robot is of a bipedal humanoid type, another important factor to consider is the degree that the operator is feeling embodied in the robot and actually present in its location (telepresence). The ROTOR VR project implemented a new teleoperation method using a humanoid robot with the aim of creating an effective control system, while also enabling the operator to feel embodied in the robot.

The goal of the system is to allow a user that is unfamiliar with operating robots to effectively transfer their movements to it, while achieving a degree of embodiment and telepresence. The experience should not be discomforting, whilst also providing freedom for the user to move unrestricted, without damaging the actual robot.

The project has utilized a stationary humanoid robot called Robothespian and on off-the-self virtual reality Headset. A framework was designed and implemented that connected the robot to the virtual avatar that is controlled by the user. These were combined with a 360° camera in order to provide the user with unrestricted view of his surrounding environment.

To test the effectiveness of the ROTOR VR project, a user study was conducted. To represent the average tasks that a stationary humanoid robot can perform, a series of tests that involve picking up objects and matching shapes using the user's depth perception. To measure the degree of embodiment and telepresence that was achieved, a small conversation occurred between the experimenter and the operator through the robot, and an experiment that observed the intuitive responses of the participant took place afterwards. To evaluate their experience a questionnaire was used, along with a short discussion regarding the experiment in the end.

Overall, the results of the studies prove that it is possible for an operator to feel embodied in a humanoid robot and feel preset at its location. Various factors that may affect this embodiment illusion are presented, like the robotic appearance of the body or the limitation of the robotic movements compared to those of a normal human.

Contents

List of Figures

Chapter 1

Introduction

In recent years there has been a significant increase in the use of remotely operated (teleoperated) robotic systems in a range of industries, like in health-care or military. Whether it is a doctor performing surgery by operating a remote surgical robot or a bomb disposal expert using a remote bomb disposal unit to disarm and dispose of an active bomb, we can trace examples of robotization in every industry. Remote operated units are not a new concept, as United Kingdom's Royal Navy used ”Cutlet”, a remotely operated submersible, to recover practice torpedoes and mines in 1970 [1]. However, developments in the fields of automation and robotics have enabled us to design frameworks and interfaces that allow operators to truly work alongside their robotic units instead of merely issuing orders. They have also made possible the conception of new applications for telerobotics, like telepresence and video conferencing or even enhancements over the current ones, space and deep-sea exploration and telemedicine. But, in order to facilitate most of these applications, advancements were required in another technology, that of virtual reality (VR).

Although the term ”virtual reality” might seem like a recent one, its concept has been around for many decades. The earliest occurrence of this term was in 1958 in an English translation of Antonin Artaud's work titled *La Réalité Virtuelle* in 1938. In 1968, an extremely heavy head-mounted display (HMD) attached to a suspension device and connected to a computer was introduced that enabled the wearer to see a virtual world [2]. This was the state of VR during Cutlet's days, which, unsurprisingly, was not able to be utilized by teleoperation. However, in recent years there have been many advances in virtual reality technologies, starting with the development of the Oculus Rift prototype in 2010 and resulting in a large number of high fidelity and resolution HMDs available to consumers today. Anyone today can pick up a VR HMD and its assorted accessories, like trackers or controllers, and be able to delve into a virtual world. So, what would happen if we combine the established archaic interfaces for teleoperating robots with virtual reality technologies?

The goal of Project: ROTOR VR is to answer that question by creating a software framework and a working apparatus that facilitates the remote operation of the stationary bipedal humanoid robot ”Robothespian”[3] through a virtual reality interface. Afterwards, the effectiveness of this setup is investigated and evaluated through a user study that includes a set of tasks to be performed while operating the robot.

Compared with the project plan (See Appendix F), the project exceeded the expected scope. The original scope of the project was about transferring the movements of the operator to the robot. However, in the end they were also able to grasp objects. Additionally, there was only one user study planned initially, but due to the small number of participants it was treated as a pilot, and the final one occurred afterwards. 4 users participated in the pilot study, which helped shape and improve the existing protocol for the final one with 20 participants over 4 days. However, there were also issues that arose with the robot's hardware, which was accounted for in the project plan as ”Probable service downtime”. Indeed, the robot had to be sent back to its creators for service after it suffered total mechanical breakdown. There were also some hardware issues during the experiments, like the right hand being unable to grasp objects, which were taken into account during the design of the experiments.

This report is organized as follows: Chapter 2 provides relevant background knowledge based on the current state of the field. Chapter 3 discusses the design decisions and creation process of the operating framework. Chapter 4 focuses on the design of the experiments, including the aim of the user study, the reasoning for selecting picking objects, matching shapes and the ”Thinking on Your Feet” approach, and the expected outcome. A detailed description of how the user study was conducted is followed in Chapter 5 and the analysis of the corresponding results is followed in Chapter 6. Conclusion, evaluation and potential further work are presented in Chapter 7. The appendix section includes the system manual, user manual, raw test results, the previously submitted Project Plan, the two versions of the protocol employed in the user study, the evaluation questionnaire, the information sheet, and the participant consent form.

Chapter 2

Background

The merging of virtual reality and teleoperating technologies has been studied in recent years. However, due to the wide range of applications in these fields, the focal point of most of the relevant research is studying and enhancing the teleoperation capabilities of remotely operated vehicles (ROVs), since they can find applications and are integral to various fields, like exploring hazardous environments - like deep space - which are unsafe for humans to traverse. Another focal point of research is improving medical remote operating units, which allow trained professionals to increase the chances of success of previously improbable surgeries. However, it is widely believed that remotely operated robots will become a common occurrence, from the workplace to leisure activities. NASA has been actively developing ”Robonaut” since 2010, a remotely operated astronaut which is a step towards combining the experience and the acuity of the human teleoperator with the expandability and durability of the robotic body [4]. Thus, further updated studies on the effects and performance of teleoperation in bipedal humanoid robots will become necessary in the coming years. Below I am going to review every proceeding, conference paper and article that I found relevant to ”Project: ROTOR”.

In order to better grasp the subject of this dissertation, it is important to first explain the concept of body ownership transfer. In 2003 Armel and Ramachandran explored the causes and effects of the - then - recently reported ”Rubber Hand Illusion” by conduction two experiments [5]. This illusion effect and this paper’s research and findings have been the basis and a point of reference for the vast majority of research regarding body ownership transfer. For their first experiment, the user sat in front of a table that had a fake rubber hand placed on it. They placed their real hand behind a vertical partition that occluded their view of it, while being able to clearly see the fake hand 2.1. Then, the experimenters applied synchronous tactile feedback, such as strokes or taps, to both hands. As a result, the user experienced the uncanny illusion that the feedback was felt from the location of the fake hand, instead of the real hidden one. The experimenters interpreted this as an example of the tolerance of the brain for discrepancy between kinaesthesia - the sense of self-movement and body position - and vision, with vision proving to be dominant in most situations. What is even more surprising, is the outcome of their second experiment. They kept a similar setup, but instead of using a fake hand, they simply applied the tactile feedback to the table. Surprisingly, the subjects experienced sensations from the table's surface, which bears no semblance to a hand. They concluded that the so-called body image, despite its appearance of durability and permanence, is a transitory mental construct that can be drastically altered by the stimulus contingencies that one encounters. The mechanisms of perception are mainly used to extract statistical correlations from the world to create a model for the mind that is temporally useful. Thus, it is possible for someone to transfer their body ownership to an external object if enough stimuli is given.

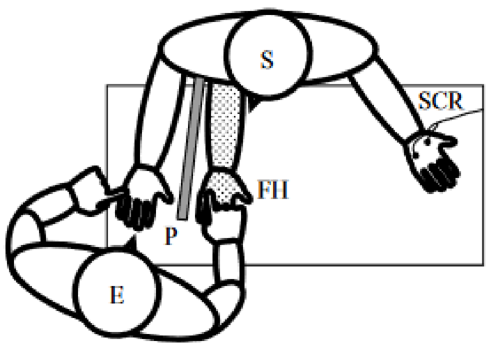

Figure 2.1: The experiment setup from [5]. The fake rubber hand is depicted as the dotted one.

Furthermore, the concept of transferring one's body ownership to a teleoperated robot was studied meticulously in 2012 by a Japanese research team of the Hiroshi Ishiguro Laboratory located in Kyoto, Japan [6]. By using "Geminoid"

(Figure 2.2) - a human-like teleoperated robot they developed - they tried to examine how the appearance and behavior of the robot will affect people that are communicating with or controlling it.

Figure 2.2: Geminoid HI-1 (left) with its source person (right) [6].

From early studies, they found that some operators would feel that when people touched the teleoperated robot it was as if they themselves have been touched, despite the lack of tactile feedback components. They hypothesized that this illusion of body ownership transfer to the teleoperated robot occurs due to the operator's observation of the Geminoid moving in sync with him. Afterward, they conducted a user study to determine the validity of their hypothesis. They enhanced their initial hypothesis by adding transfer delays between the operation of the android and the visual feedback of its motion. So, they ended up with three conditions for the experiment, one where the movement is perfectly synchronized, one where they added a slight (one second) delay and one where the robot did not move at all. Based on their hypotheses, they made two predictions, that the participants will show larger responses in the first two conditions compared to the third one. Thus, they conducted a total of six trials, two stimuli - bending a robot's finger and injecting the robot's hand with a syringe (Figure 2.3) - for each condition per participant and verified whether body ownership transfer had occurred. Even though the "sync hypothesis" was confirmed without a doubt, the "delayed hypothesis" was not, as the results were inconclusive. This adds to the point that delays, even minor ones, degrade the teleoperation experience drastically if not dealt with.

The closest research of interest regarding this dissertation is the PhD thesis of

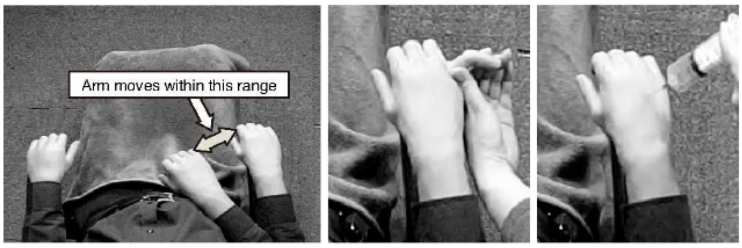

Figure 2.3: Simulated participant's view in user study from [6]: (left) normal view showing arm movement range; (middle) finger bending stimulus; (right) injection stimulus.

Sameer Kishore from the University of Barcelona[7]. Similar to this study, the author tries to apply the phenomenon of full body ownership illusions in the context of humanoid robots. By using previously researched and widely accepted methods of embodiment, he tries to explore whether they can be applied to a humanoid robotic body or not. Additionally, he also developed a framework that would allow an operator to remotely control a robot. To test the functionality of his system and to determine the factors that affect body ownership transfer to robots, the author carried out two experimental studies and one case study. The robots that were used were a modified version of Robothespian, albeit placed in a moving platform to enable movement, and a Nao robot, which was only used in the third study.

[a]  [b] 

Figure 2.4: The two full-body tracking systems that were used to control the robots in study [7]. The suit shown in (a) was used for the system described in the first two case studies, while (b) was used in the final one.

His first study was motivated by developing a system that would allow disabled patients to easily operate robots and compared two systems of control, one based on eye-tracking and a Steady-State Visually Evoked Potential (SSVEP) based Brain-Computer Interface.

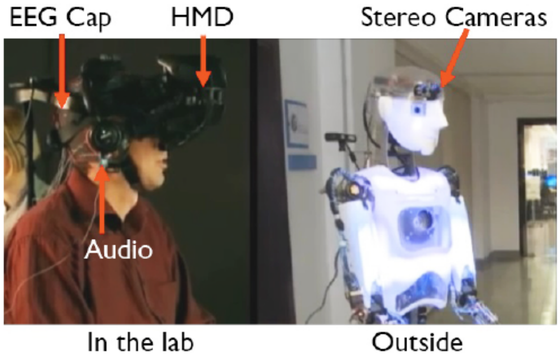

Figure 2.5: Experimental setup in study [7]: on the left, the participant in the lab, fit with the HMD and the EEG cap (only for the SSVEP condition) and the headset for communication; on the right, the robot in the outside area with the stereo camera.

His second study utilized the same control systems as before, with the added ability of moving the robot around using its attached mobile platform. The motivation of this was to determine whether giving the participant more control over the robotic body would result in higher degree of embodiment. In the final case study, he utilized both robots in order to project the operator in three places at once. This investigated whether he could cope with being in three places at the same time, while embodied in three different bodies. His results indicate that it is possible to facilitate body ownership transfer between a human and a humanoid robot. He also analyzes factors that can affect this facilitation, like the appearance of the robotic body. Overall, this paper is the basis of my thesis, as in the current paper a more accessible controlling system is proposed and the factors that may affect body ownership transfer and displacement illusion are further investigated.

Another paper of interest was presented at the Information Visualization conference in 2002 and explores the subject of cooperative robot teleoperation through virtual reality interfaces [8]. Albeit old, this peer reviewed research explores the topic of teleoperation so deeply and fully that it has been referenced by many relevant papers, such as ”Application of Augmented Reality in Mobile Robot Teleoperation” in 2018 [9]. It provides an analytic explanation of the complexities of collaborative teleoperation and defines some guidelines in designing an ideal user interface utilizing virtual reality desktop. As we can see from Figure 2.7 it is obvi-

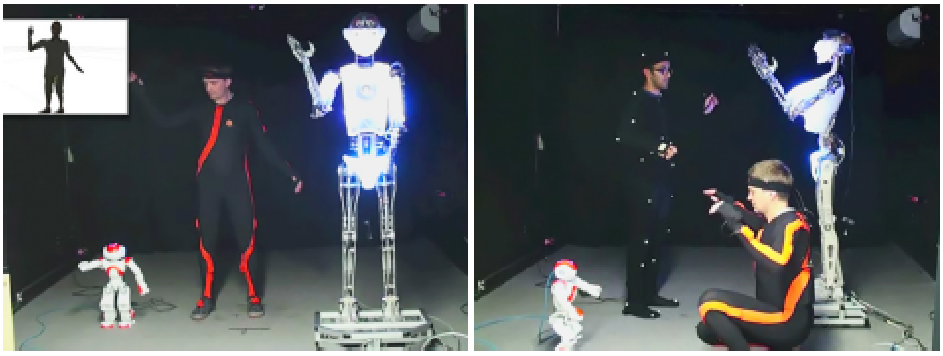

Figure 2.6: Examples of combinations of tracking systems for controlling humanoid robots in study [7]. The participant on the left image controls both the Nao and Robothespian, and a virtual avatar. The right image shows an example where two participants physically interact with each others humanoid robot representations. One participant controls the Nao via the Optitrack system, while the other participant controls the Robothespian with the other system.

ous that the technology utilized by its authors is very outdated. However, this paper was not selected for its technical part.

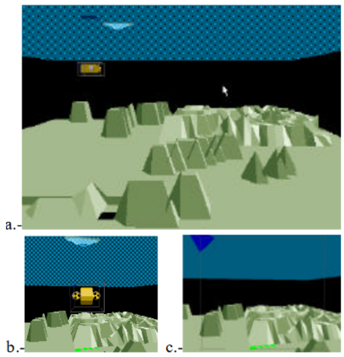

Figure 2.7: Different viewpoints from a virtual world, using the interface developed in the project from [8].

The basic principles and guidelines the authors suggest are still relevant, like presenting details of the position information and spatial orientation information on the interface, and I aim to incorporate them into my project. They presented two types of interfaces, one with an emphasis on information visualization and one with an emphasis in improving the user's control (Figure 2.8). These suggestions contributed heavily to the evolution of teleoperation interface designs, as is evident from the similarities to present day examples.

The next relevant paper was published in 2007 as part of the *Systems, Man, and*

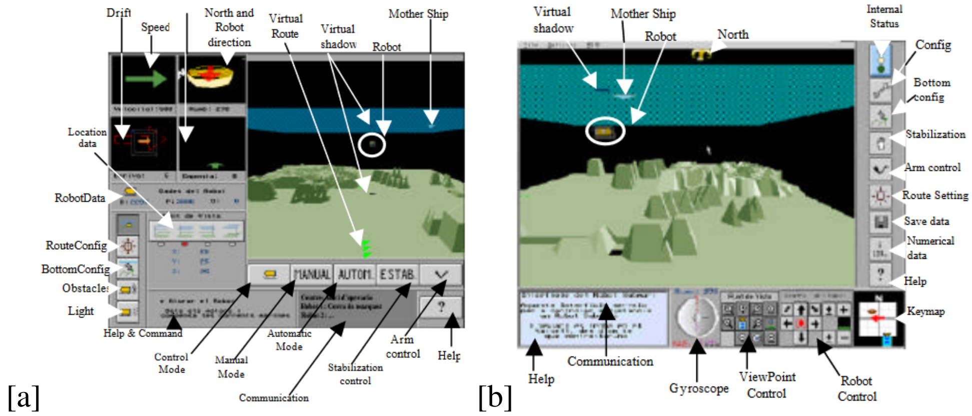

Figure 2.8: (a): Interface for teleoperation with emphasis in information visualization.
(b): Interface for teleoperation with emphasis in controllability.

Cybernetics journal [10] and is exploring human performance issues and user interface design for teleoperated robots. The authors predicted that in the future humans working alongside robotic systems in order to accomplish forbidding or impossible tasks would be a common occurrence. Truly, there is an increasing amount of tasks where teleoperation is presently required to complete them, like NASA's aforementioned Robonaut project. They identify the human factor as the most challenging component of teleoperation. Being remotely located, the operator might cause the effectiveness of the mission to suffer due to compromises to his situational awareness of the remote environment. Therefore, the authors examined more than 150 papers that cover human performance issues related to technical factors, like time lags - the delay between input action and (visible) output response -, video image transfer bandwidth or two-dimensional views and suggested mitigation solutions. While most of the issues are related to outdated technology limitations, like limited bandwidth, there are some that may apply to today's teleoperation methods too, like time lags. Even though transfer rates have vastly improved over the years, as well as the technology in general, some issues cannot be eliminated due to their nature. For example, time lags still exist and cannot be eliminated for teleoperation over long distances, since speed of light issues come into play. To transfer data halfway around the Earth via a fibre-optic connection takes 100ms, without including relay delays. While they are not as bad as before - right now our average latency in the video stream is five to ten milliseconds, whereas at the papers' time it could range from 170 milliseconds to two seconds - they can still impact the quality of the teleoperation process. These time delays have been associated with motion sickness, which can be caused by cue conflicts, meaning discrepancies between visual and vestibular systems. The paper's suggested solution for this issue is the implementation of ”predictive displays”, a concept that is applicable even today. By using the teleoperators movement inputs, a simulation of the kinematics execution is being shown to him with no delay. This way the robot can execute the movement, while the operator is calculating his next move. Thus, it is interesting - and surprising - to see that some of the suggested troubleshooting solutions can be applied today, albeit in a modern fashion.

Another paper that is relevant to this study is an analysis of the aforementioned issue of time delays and was presented in the 2010 International Conference on Virtual Environments, Human-Computer Interfaces, and Measurement Systems [11]. It details the usage of virtual reality in improving teleoperation by using a simulated environment in parallel to the real system in order to improve the quality of the remote control. The suggested proposal in dealing with network delays is meant to be applicable to various fields, such as industrial applications, measurement systems and - relevant to us - robotics. The authors' approach is a more developed version of the ”predictive displays” method that was outlined before, for which they used the Atelier de Realite Virtuelle (AReVi) framework - which is translated to Virtual Reality Toolkit - and its module ”Physics” [12]. While depreciated by the tools available today, AReVi was a framework developed in 1998 that made it possible to create virtual reality applications with minimal programming effort. Its module (”ArPhysics”) was aimed to be a high performance simulating rigid body kinematics library that utilized an advanced physics engine. The authors created a program that used a test environment to simulate a users control inputs to a 5 Degrees Of Freedom (DOF) robotic arm (Figure 2.9) with delay, which represents the time delays encountered in a network. The amount of millisecond of simulated delays was set by the authors on different values during their experiment.

To counter that delay, the orders sent to the real robot will also be transmitted

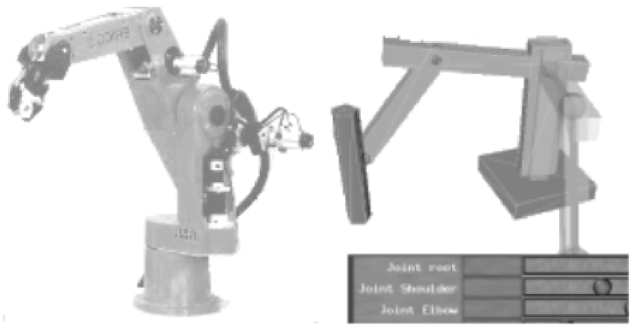

Figure 2.9: Real (left) and virtual (right) robotic arm used in project [11].

to a virtual one. This way ”ArPhysics” can simulate the behavior and the kinematics of the real robot by the end of the input, and move it accordingly. During this, the operator will be shown the virtual robot which will move at a reduced speed, in order to allow the real one to synchronize. In the end, the user will be unable to perceive any discernible latency issues in his commands. This method served as the basic concept for my project’s implementation, which will be detailed in the next section.

Finally, an interesting approach to evaluating a person’s embodiment experience is present with the form of a questionnaire in a paper published in 2018 [13]. The focus of the paper is about creating guidelines and a basic set of questions that can be used to measure the degree that a person feels embodied in a virtual avatar. The authors review the methods, especially the questionnaires, that have been used in past experiments that revolved around the phenomena of embodiment and telepresence. They included an overview of previous experiments that have included both qualitative and quantitative metrics to measure the embodiment of avatars, such as questionnaires, heart-rate monitors, skin-conductance, and electroencephalogram (EEG). Their findings suggest that the extent to which the illusion of embodiment is experienced varies between participants, despite being strongly associated with the performance and reactions inside virtual reality. So, after they perform that review, they propose a standardized embodiment questionnaire based on 25 questions, some of which are used in the questionnaire of this study.

Chapter 3

Implementation

To evaluate my approach on facilitating the body ownership transfer illusion, a control system had to be developed that allows a user to control a humanoid robot by only using an off the self virtual reality headset and controllers. In addition to that a visual system linking the robot to the operator had to be implemented, as the robot does not have a sufficient one.

3.1 Hardware

In this section the design and construction process of the visual system is being analyzed, as well as any other integral hardware part of our experiment, like the capabilities of the robot.

3.1.1 Robothespian

The humanoid robot that is going to be used in this project is ”Robothespian” (Figure 3.1), a humanoid robot designed for human interaction in a public environment created by Engineered Arts [14]. This makes it ideal for our project's scope since a human operator will find it easier to adjust to its humanoid body movements, thus making it possible to achieve a degree of body ownership transfer - making the teleoperator feel as if the robotic body is an extension of his own. There are, however, limitations to the robot's movements that do not apply to an average human, like the fact that it can only extend its arms sideways approximately 45°. These limitations, along with the fact that its robotic appearance might dissuade people from identifying as it, serve as excellent points of interest in this study. Robothespian moves its body parts using a hybrid actuation system, meaning a series of elastic motor drives running in parallel with pneumatic muscles, which offers a compliant lightweight design with high strength to weight ratio. Unfortunately, the one owned by our lab was an old, custom model that has very temperamental and unreliable motion servos that tend to unexpectedly malfunction. This has been taken into account during the design of the user study, and it will be further discussed in Chapter 4. Finally, Robothespian operational system is the ”Tritium” framework, a proprietary Python-based software developed by Engineered Arts.

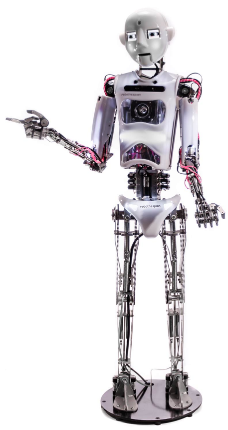

Figure 3.1: Robothespian model RT3 by Engineered Arts [14].

3.1.2 Visual System

The biggest contribution to providing the operator with a real-time, immersive virtual reality experience comes from the video feed. A live video feed of the robot's perspective is required, with as little input lag as possible in order to make the operator feel actually present at the remote environment. Robothespian's hardware, however, is unable to facilitate this due to the lack of an embedded virtual reality capable camera. The model has a single camera that is used to provide an simple feed of the robot's field of view. It has low resolution and refresh rate and, while it is adequate for its purpose, it cannot be utilized to create a virtual reality feed. This feed has to provide the operator with an view of his surrounding that is in high resolution and as close to real time as possible. Thus, the camera must be capable of high resolution output (at least 2K) and have a fast refresh rate (at least 30Hz).

In order to create this type of feed, two specific camera systems are widely used - stereoscopic cameras and 360° ones.

3.1.2.1 Stereoscopic Cameras

Stereoscopic or stereo cameras are usually attached on a mechanical platform that allow the user to move it according to his viewpoint. By using two separate lenses at the same time, two different vantage points are being utilized when shooting in stereo, one for each eye, allowing the viewer to determine depth accurately, thus creating a hyper real virtual experience.

Figure 3.2: Example of stereoscopic content [15].

3.1.2.2 360° Cameras

A standard 360 video is just a flat equirectangular video projected on the inside surface of a sphere. The central view point, which is the user's head, is on the inner part of the sphere looking at the inner surface, where the video is displayed. So, while it removes the need for a mechanical moving platform to attach the camera to, it degrades the viewer's perception of depth.

3.1.2.3 Final Camera

While it seems like a stereoscopic camera system is superior - and it is under certain conditions - the final camera is a 360° one. That might come as a surprise, especially after seeing that stereo cameras were used in Kishore's Robothespian robot [7], as seen on Figure 2.5. Initially, I decided to recreate a similar visual setup, by using one of UCL VR Lab's stereoscopic cameras. Unfortunately, the difference

Figure 3.3: Example of monoscopic content [15].

between the aforementioned Robothespian model and our own became apparent, since the movements of the unit's head experienced input lag and caused tremors to the cameras which resulted in an unstable, shaky and delayed video feed. So, in the end, I decided against using that type of visual feed system as the experience was discomforting and would be detrimental to the experiment. Of course, one with sufficient mechanical skills and budget could either create or buy a stereoscopic camera platform that significantly reduces the tremors.

To select an ideal 360° camera for this specific teleoperation project, apart from the aforementioned requirements of high resolution and fast refresh rate, it must have a suitable size to mount on the robot's face plate, PC connectivity and be capable of real-time streaming protocol (RTSP) live-stream. Taking these criteria into account, the best present choice was the "RICOH THETA V" (Figure 3.4) [16]. It is capable of 4K streaming at 30Hz, or 2K streaming at 60Hz via a direct connection to the computer, all while being very compact and affordable.

In order to create a VR feed from a 360° camera, the video output is projected on the inside of a sphere (Figure 3.5) and the operator's virtual viewpoint is placed on its center. By moving their head around, the operator is shown the appropriate section of the spherical projection without the need for the robot to also move its head accordingly. This way they can have an accurate, real-time panoramic view around the robot while the camera is attached to the stationary head, which minimizes the lag input and the tremors caused by moving the camera around.

Figure 3.4: Ricoh Theta V camera [16].

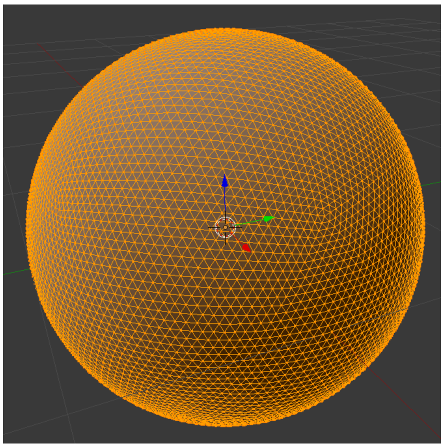

Figure 3.5: Video projection sphere. The video is projected on its inner surface, and the operator's viewpoint is centered.

Finally, to enhance the degree of body ownership transfer, it is important to attach the camera directly on the robot's face-plate, at the approximate height of its eyes. While the robot does not have any need to move it's head, the operator's point of reference (the lens) must be anatomically correct in respect to the robot's viewpoint (between its eyes). This is facilitated by attaching the camera on a mask worn by the robot (Figure 3.6).

3.1.3 Virtual Reality Headset

The last hardware component that is used in this experiment is the virtual reality headset and its accessories. One aspect of this project is the creation of a teleoperating control system for a bipedal robot that does not require a specialized

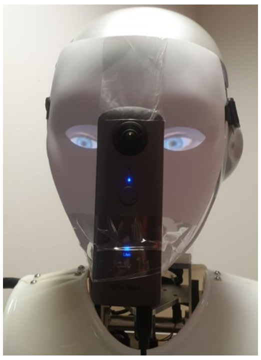

Figure 3.6: 360° camera attached to the mask worn by the robot.

interface or a full body tracking suit, like that in Figure 2.4. Since the system is developed by using SteamVR as its virtual reality operation framework, it is effectively compatible with the majority of virtual reality headsets and controllers in the market. For the this experiment HTC Vive pro (Figure 3.7) [17] was used.

Figure 3.7: HTC Vive Pro pack [17].

3.2 Software

After having assembled the basic aforementioned hardware components, the next step is to create the teleoperating control system that will enable the user to remotely operate Robothespian. As mentioned before, Robothespian's operating system is the proprietary Python-based framework ”Tritium”, with which our controlling system has to interface via a User Datagram Protocol (UDP) connection. UDP is used primarily for establishing low-latency and loss-tolerating connections between applications (process-to-process communication), whereas the alternative Transmission Control Protocol (TCP) facilitates host-to-host communication. So, creating both a Unity scene that sends limb movement commands and a Tritium control function that reads and executes them is required.

3.2.1 Tritium Control Node

The custom programmable Tritium functions that let the user operate Robothespian are called ”Nodes”. By programming them, the user has the capability of controlling each moving part of Robothespian. In this node, a UDP server is being initialized and is listening for commands to execute. The commands are sent as strings, each following a naming protocol - each string starts with the code for the limb name, followed by signed values for each of its movement dimension. For example, the command: ”LArm:-10-20+30” would indicate that the left arm has to move -10 degrees up from its starting point (pitch), -20 degrees outwards (roll) and it should twist +30 degrees (yaw). The validity of each movement is handled in the Unity scene. Finally, it can receive a command to open or close each fist and return to its resting pose. Complete code for this node can be found in the Appendix E.1.

3.2.2 Unity Control System

The control of Robothespian’s movements is facilitated by creating a correlation between the real robot and a virtual, operator-controlled avatar. While it would be simpler to just transfer the operator’s raw movements directly to the robot, that could result in hardware damage to it by executing inappropriate movements, like crashing the arms together. Instead, by having the user directly control a virtual robot in a regulated virtual environment, limitations and constraints akin to its specifications can be easily set. The development engine chosen for the creation of this virtual environment is Unity since it provides a ready-to-use game engine and flexible real-time tools. Its popularity and extensive documentation provide easily found solutions for issues that may arise. The compatibility and easy integration with the aforementioned SteamVR is another great incentive. SteamVR is an API that all the popular PC VR headsets can connect to and its Unity Plugin handles many useful VR development tasks, like loading 3d models for VR controllers, handling input from those controllers, and providing an interaction framework for them with the rest of the scene's objects [18].

3.2.2.1 Creating the Virtual Avatar

The creation of the aforementioned virtual operator-controlled avatar requires several steps. First of all, in order to create the avatar itself a 3D model has to be used. There is a wide range of free models available at Unity's Asset Store that fit this project's needs, like realistic ones (Bodyguards (Figure 3.8.a) [19]), cartoonish ones (Cyborg Character (Figure 3.8.b) [20]) and robotic ones (Space Robot Kyle (Figure 3.8.c) [21]).

Figure 3.8: Character avatar models (a): Bodyguards (b): Cyborg (c): Space Robot Kyle.

Artistic choice aside, to determine which model would be used for the avatar, compatibility with the Inverse Kinematics (IK) solution had to be considered. Since most current off-the-shelf VR Headsets do not include trackers for every human joint, like elbows, the need to extrapolate their position in order to have an accurate moving virtual avatar and experience arises. Inverse Kinematics is the mathematical process of recovering the movements of an object in the world from some data. In this project this data is the geometry that describes the position and orientation of a characters joints. A dynamic connection between the parts of the avatar is created and their position is calculated. For example, the movement of the non-tracked elbows can be approximated based on the position of the tracked head and hands. The two dominant IK systems that are Unity and VR compatible are, at present, Unity's own default Mecanim IK and RootMotion's Final IK [22]. In order to determine which solution was better, a Bodyguard avatar had its bones rigged according to each system's specification. After experimenting extensively with both, Final IK was chosen as Mecanim was very limited and had a lot of drawbacks. For example, Mecanim's lack of a handle for the head kinematics resulted in situations where the head was not perfectly aligned with the camera, meaning that when turning you could occasionally see the head from the outside or the character model would bend itself unnaturally and flip itself inward when looking in a particular direction and holding the controllers at a specific angle. In addition to that, the elbow positions were not calculated accurately, resulting in great disparities of real and virtual arm movements. On the other, Final IK's VR capabilities proved superior as there were no issues of the model folding or the virtual avatar not matching approximately the operator's movements. Finally, in order to prevent damage to the robot, colliders (forbidding boundaries) were added to each body part that prevented each part from hitting or intersecting with the rest of the body.

3.2.2.2 Calibrating the Virtual Avatar

During pilot testing the accuracy of the virtual avatar's movements with members of UCL's VR Lab, the need to calibrate the avatar's size based on the operator became apparent. While the avatar's movements were accurate for operators of average height, they were inaccurate for taller users, as their body moved beyond the limits of the virtual one. To counteract this, a custom script was written that assumed the user was holding his controllers fully extended to his sides and after providing the assigned key input, adjusted the avatar's size procedurally until it matches the user's arm length estimation. This pose is called T-Pose or Reference Pose (Figure 3.9) and is a default unanimated state of some models in 3D graphics. The code for this can be found in the script "Calibration.cs" attached to the Calibration Module in the scene and in the Scripts folder in the project's repository. After trying out all three potential character models, the one that had the least disfigurement and behaved best through any re-sizing was the Bodyguard, and was selected as the one used for this project.

Figure 3.9: T-Pose

3.2.2.3 Robothespian Movement Commands Creation

After having created a fully working and adjusted virtual avatar, the next step was to extrapolate commands that can be sent to and executed by Robothespian. Robothespian moves each part based on certain degree values and they can be seen in Table 3.1.

Body Part	Pitch Min	Pitch Max	Roll Min	Roll Max	Yaw Min	Yaw Max
Torso	-15	15	-15	15	-15	15
Upper Arm	-90	0	-90	-30	-30	40
Forearm	-80	10	N/A	N/A	N/A	N/A
Wrist	-15	15	-90	90	N/A	N/A

Table 3.1: Robothespian Movement Range.

In order to stay on the safe side, the aim of the mapping that was to operate Robothespian within a few degrees off the extreme values and can be seen in Table 3.2.

Command	Body Part	Pitch Min	Pitch Max	Roll Min	Roll Max	Yaw Min	Yaw Max
Torso	Torso	-10	10	-10	10	-10	10
LArm	Left Upper Arm	-80	-10	-85	-35	-25	35
LForearm	Left Forearm	-75	5	N/A	N/A	N/A	N/A
LHand	Left Wrist	-10	10	-85	85	N/A	N/A
RArm	Right Upper Arm	-80	-10	-85	-35	-25	35
RForearm	Right Forearm	-75	5	N/A	N/A	N/A	N/A
RHand	Right Wrist	-10	10	-85	85	N/A	N/A

Table 3.2: Robothespian Movement Commands Range.

The way Unity stores the rotational values of each limb in its models, initially, seemed to be sufficient to facilitate the mapping to a degree. After some testing of the ranges of the movements of the avatar and the correlation to those observed on the robot, an initial mapping was made. For example, a rotational value of -10 on the Z-axis of the avatar's torso would match the rotation of Robothespian when given a rotational value of 10 to its torso pitch. Another example, where the mapping was not that obvious, is that of the upper arm, where its initial position of 0 degrees on the Z-axis matches that of Robothespian's at -80 value pitch and at 35 on the Z-axis that of -10. So, the current Robothespian command's value could be found by doubling the avatar's Z-axis angle and subtracting 80 degrees.

After having created the mapping that enables the calculation of the values Robothespian has to receive in order to match the operator's movements through the virtual avatar, Unity Engine's Networking and Socket API was utilized. This allowed the establishment of the aforementioned UDP connection, achieving bidirectional communication between Unity and Tritium, effectively synchronizing the movements of the two robots. Unity's code of that connection can be found at the "UDPSend.cs" script and Tritium's at the Appendix E.2. The commands that are being constantly sent follow the protocol explained at 3.2.1 and are comprised of the command parts of Table 3.2 and the values that result from applying the mapping.

However, after initial testing within the Lab, an unexpected issue with the mapping was found. While in theory there seemed to only be one possible combination of degrees that correspond to the position of a limb, in practice it became obvious that this was not the case. Due to how Unity initializes the positioning of the object on each axis, the initial point of reference for the rotational transformations may be initialized differently each time. This can be seen in Figure 3.10, where we can have two different degree combinations ((85, 24, 78) and (85, -35, 20)) that result in very close movements. To further explain this issue, one must explain how Unity resolves rotational degrees. Unity internally uses Quaternions to represent all rotations. They are a number system that extends the complex numbers and are used in particular for calculations involving three-dimensional rotations such as in three-dimensional computer graphics. They are compact, don't suffer from "gimbal lock" and can easily be interpolated [23]. Gimbal lock is the situation where two axes are locked in the same attitude providing the same rotation. For example, when playing a first person shooter and one points their weapon vertically at 90 degrees, they are no longer able to point the weapon left or right, as moving the mouse sideways would only rotate the camera alongside the weapon's axis. However, quaternions are based on complex numbers and are not easy to understand intuitively. To present them to the user in an simple and understandable format, quaternions are transformed into Euler angles, the most common form of angle presentation. Unfortunately, due to the nature of the transform to and from Euler angles, two different Euler angle based rotations in the Y-axis - like in the two aforementioned degree combinations - can result in two final positions that are close to each other, like in Figure 3.10.

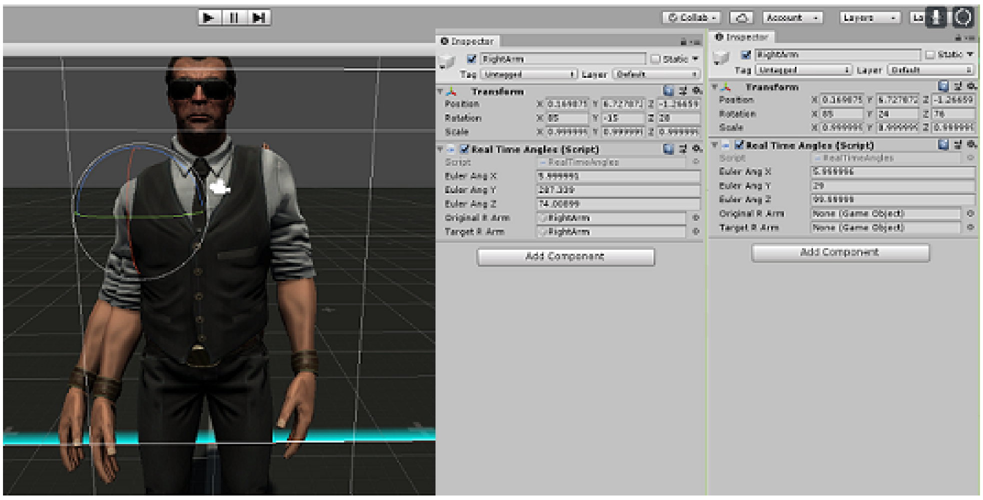

Figure 3.10: Mapping issue in Unity.

So, instead of trying to create a multiple or dynamic mapping system, which could be prone to errors due to unforeseen transformation combinations, the system's detection method was entirely redesigned. Instead of relying on the values each body part has, trackers were placed on each limb, whose relative position was calculated in regards to their initial position according to measurement planes. Small, invisible to the operator sphere objects attached to each moving limb are used as trackers for this project's virtual avatar. For each moving limb an initial tracer - that marks its starting position - and an active tracker - that marks its current position are placed. Each movement of that limb moves the active tracker, and the difference in distance on its corresponding axis result in rotational values that can be mapped to Robothespian Tritium values. An example of this can be seen in Figure 3.11, where the white sphere on the right represents the initial tracker of the right upper arm and the highlighted sphere on the left represents the active tracker. Their difference on the Z-axis (green arrow) represents the extension of the arm and can be mapped to Robothespian's right arm Roll value. To make the visualization easier, an invisible to the operator blue plane is created, where we can see the point that the horizontal extension of each tracker - purple for the initial tracker and yellow for the current tracker - intersects with it. To further help with visualization, we place a small rectangle there. The Z-axis difference of the 2 triangles is mapped to the Roll upper arm value. Additionally, since the spheres are attached to the avatar's limbs, they inherit their rotational values, which is counter productive to our system - imagine the white lines in the figure moving around alongside the arm instead of being parallel to the ground. Therefore, a script was written that counters any rotation imposed on them and its code can be seen at Appendix E.3.

3.2.2.4 Heads-Up Display Assistance

Finally, as pointed out by the research papers that were studied, careful consideration had to be given to the user interface. The operator has to have some visual aid that assists him in operating Robothespian effectively and efficiently. To that end, the module Figure_Helper was created, in which a clone of the Bodyguard is made that mimics the movements of the operator controlled Bodyguard. In addition, an extra set of limbs have been placed, that adhere to the restraints set by Robothespian and are not visible until a limb moves beyond that movement range. Then the extra set of limbs stay at that limit and become green colored, while the active limbs become red. This provides the operator with an indication that he has to bring these specific limbs back within the correct range. For example, in Figure 3.12 it can be seen that the arms are bend backwards in a position that cannot be performed by Robothespian. The same applies for the right hand.

Based on this system, the operator needs to have a complete view of the as-

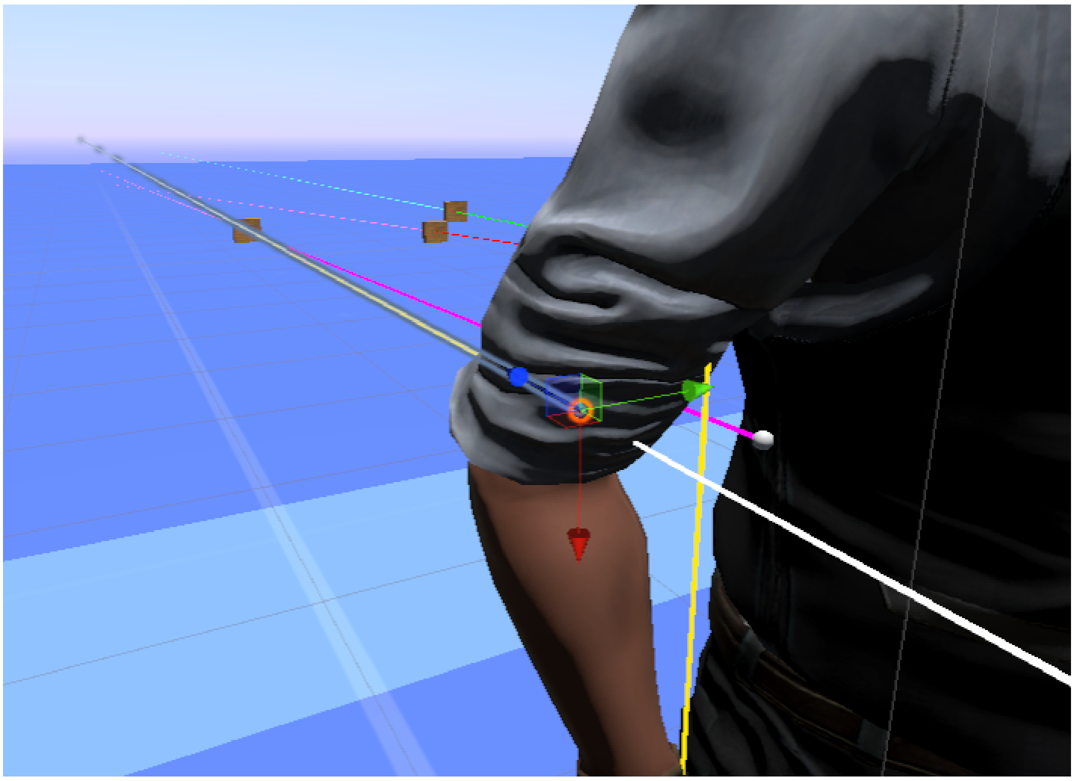

Figure 3.11: Movement tracking system. Trackers are placed on each moving limb and their distance difference on an axis can be extrapolated to rotational values for Robothespian.

sisting avatar in a way that is intuitive and can be accessed easily. To that end, four cameras are placed around the avatar (Figure 3.13) that provide the operator with views from the left, front, right and top side. Afterwards, these views are projected on the upper right corner of his virtual interface (Figure 3.13).

3.2.2.5 Scene Organisation

To make the process of enabling a user to control Robothespian as efficient as possible, one Unity scene with two phases, each based on its own module, is required. In the first phase (Figure 3.14) the avatar calibration is taking place through the already enabled upon initialization Calibration module. The script that adjusts the avatar to the operator's measurements is activated through the experimenter's input and a mirror exists that allows the user to see their avatar's reflection and judge the accuracy of its movements. The Measurement module that determines the positioning of the avatar's limbs is active in both phases. The second phase (Figure

Figure 3.12: Assistant figure providing feedback with colored limbs.

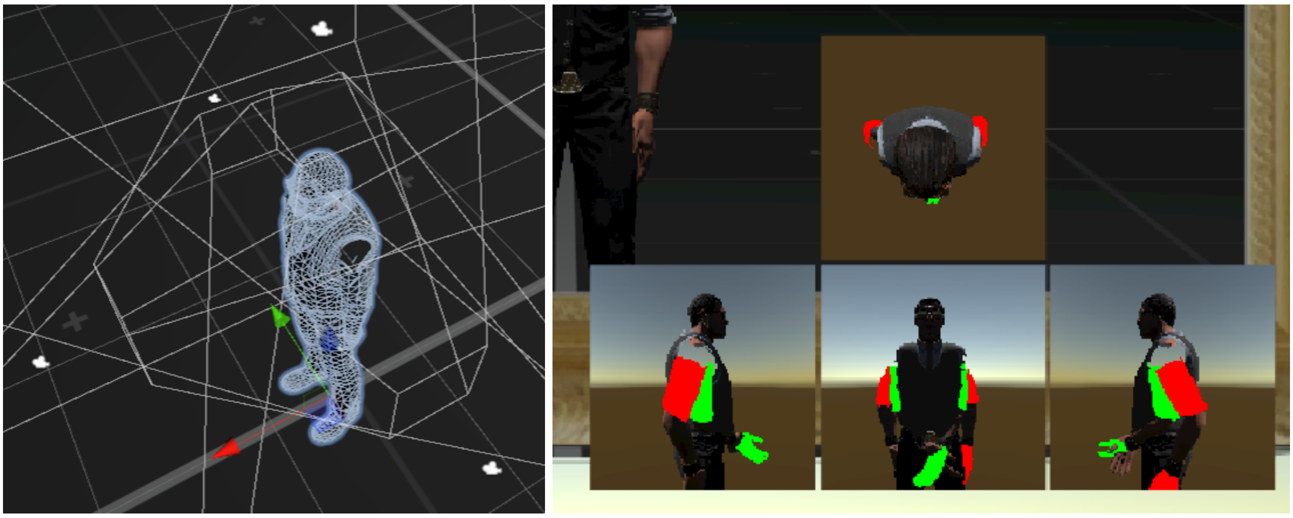

Figure 3.13: Assistant model and the cameras providing each camera view (left). Assistant model providing correctional information (right).

3.14) is activated upon experimenter's input and the robot control is taking place by enabling the module that is responsible for controlling the robot. Every visual element from the first scene - the Calibration module - is disabled (hidden), and the camera's visual feed is enabled. The user's virtual placement in the scene is in the center of the sphere upon which the camera feed is projected.

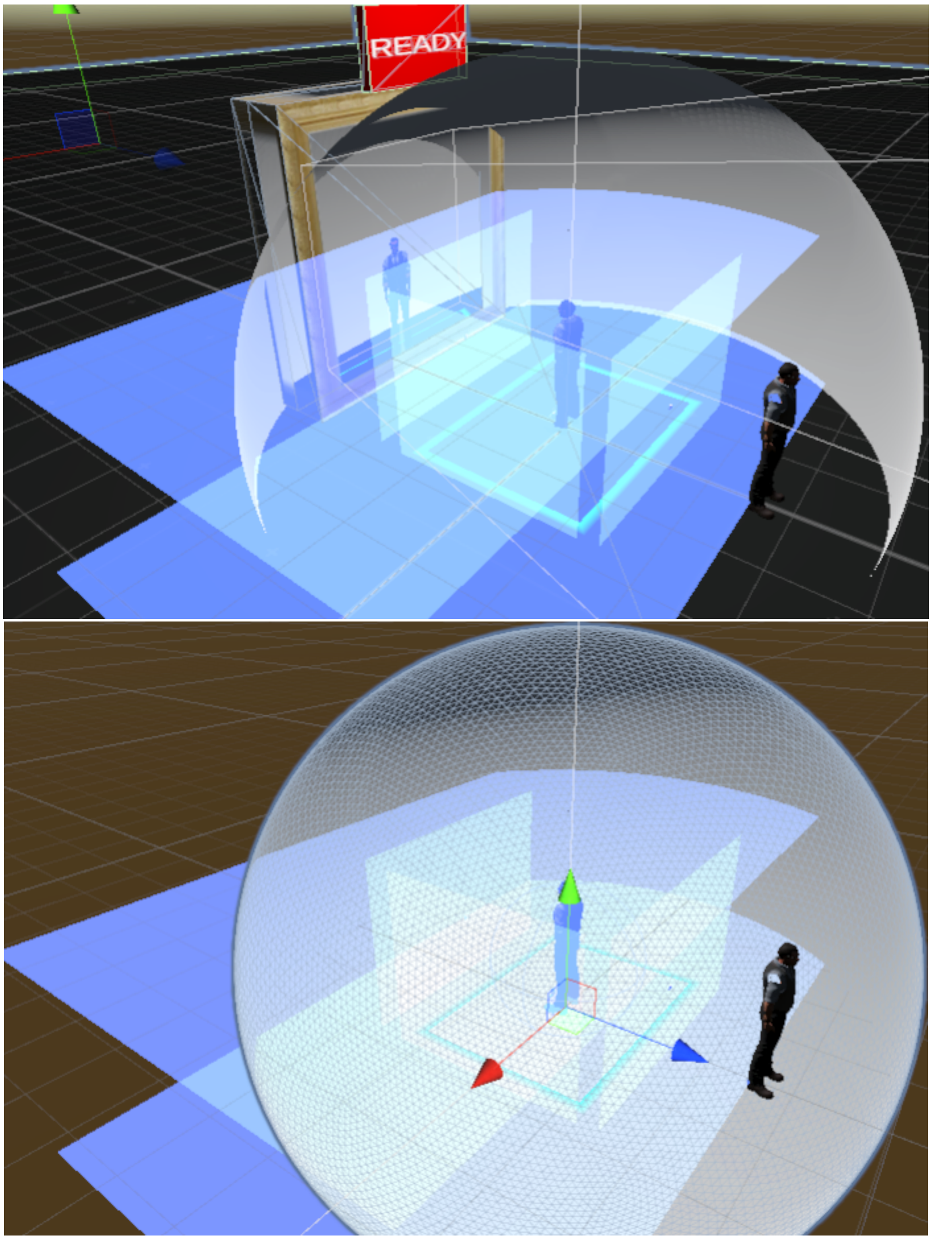

Figure 3.14: Avatar calibration phase scene view (upper).
Robothespian Teleoperation phase scene view (lower).

Chapter 4

User Study and Experiment Design

To verify the effectiveness of the teleoperating control system in facilitating the phenomena of body ownership transfer and embodiment to a remote robotic body, two user studies were designed and conducted. This chapter describes their overall structure and experiment design, while their execution and exact protocol is described in Chapter 5.

4.1 Aim and Task Selection

It became obvious that while the phenomena of telepresence in conjunction with teleoperation and telerobotics have been studied extensively, as well as those of body ownership transfer and embodiment, there exists little experimental evidence and research that combines them together. Therefore, a lot of improvisation was included in creating a control system and an effective apparatus that can allow a human operator to control the robot through a virtual reality interface, while also taking into account Kishore's research without overlapping it. To determine its effectiveness, the study needs to contain experiments which require the operator execute tasks suitable to the robot, while they are progressively experiencing the phenomenon of telepresence.

As first task, the operator is asked to pick up an object (Figure 4.1) and place it inside the adjacent box. It is a challenging task, since the object is placed in positions that can only be reached by a fully extended or retracted arm. It is also a task in which they must fully utilize the assistant HUD in order to resolve any operational issues.

As second task, the operator is handed a smaller object attached to a rectangular base (Figure 4.1) and is asked to place it in a box that has a rectangular hole on it. This task requires spatial awareness and challenges his depth perception, which is not optimal due to the aforementioned use of a 360° camera. Additionally, the box is placed in positions that follow the same principle as before.

Figure 4.1: Objects used for the experiment. On top of the box are the object for the first task on the right and the object for the second task on the left.

As third task, the operator is asked to have a conversation with the experimenter through the teleoperated robot. During this phase, they take their mind off executing mechanical tasks and by interacting with another person, they get more used to their robotic body and view. This task was not present at the pilot study, and was added due to suggestion from the pilot participants.

As final task, the operator is asked to execute commands that are issued by the experimenter verbally in fast succession, in order to make them react instinctively. As final command they are asked to locate the experimenter and they will either use their real body as point of reference for the search or the robot's location. This task is the most important to determine the degree they felt embodied and present at the location of Robothespian.

4.2 Limitations

Unfortunately, there were two major limitations that had to be taken into consideration when designing the experiments. The robot cannot be moved from its place due to connectivity issues both with its air pressure supply and its specialized network system. That means that the task execution environment has to take the limited space in account. Additionally, due to mechanical failure the torso and the finger joints of the right hand were inoperable. Since the additional repair time is forbidding to the scope of the project at this stage, the tasks took these disabilities into account too. So, the design of the tasks is focused around the left hand in a restricted space.

4.3 Structure

The experiment is separated in three stages through which every participant will go and perform exactly the same tasks.

4.3.1 Stage 1: Training

To accommodate the participants to the movements and control of Robothespian, they first have to perform a training stage.

4.3.1.1 Task

The participant is instructed to spend up to five minutes getting used to the assistant HUD and the movements of the Bodyguard avatar, without having any particular task in hand. They are instructed to move their arms around and see how the environment responds. When the participant feels confident enough, or the five minutes pass, the experimenter activates the Robothespian controls and the operator is actively in control of the robot and receives a visual feed from the mounted camera. They receive the same instructions. When they are confident enough, or the five minutes pass, the experimenter places a target in front of the robot in an easy position, so they can attempt to grab it, lift it and place it inside the box. This happens twice.

4.3.1.2 Metrics and Measurement

There are no metrics captured in this stage. Its sole function is to give the participant time to get used to the controls of the robot and the usage of the HUD.

4.3.1.3 Expected Outcome

The aim of this stage is to accommodate the participant to the avatar's and robot's controls and the HUD. Through moving their arms around, they are bound to execute many moves that cannot be done by Robothespian, so they will get to actively use the HUD to correct their movements. Also, through trying to grab the targets, they will experience the sensitivity of the robot's movements and progressively get used to their new depth perception through the camera.

4.3.2 Stage 2: Grabbing Objects

In this stage, the participant performs the aforementioned first task six times in a row.

4.3.2.1 Task

The participant has to grab the specified object and place it inside an adjacent box. If the object drops, the experimenter will reset it at the same position. The six positions are chosen based on the extreme limits of Robothespian's arm as shown in Figure 4.2. For example, the first one requires the subject to fully retract his arm, while the third one requires them to fully extend it.

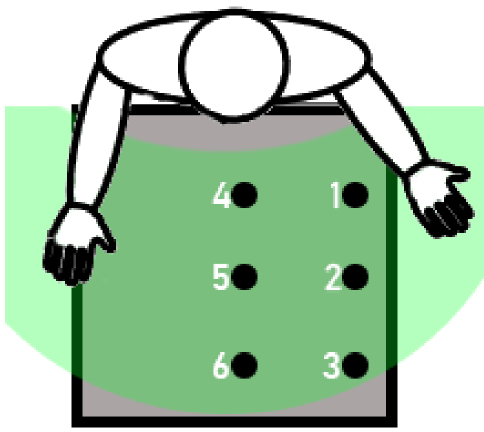

Figure 4.2: Depiction of the six placements of the objects. The green zone is the effective range of the arms and the black circles represent the positions.

4.3.2.2 Metrics and Measurement

For each trial, a number of metrics are recorded. These metrics are the time from the first movement of Robothespian's arm to the point where the object is placed inside the box and the number of times the object had to be reset due to the participant knocking it down.

4.3.2.3 Expected Outcome

We have two hypotheses regarding this stage:

- **H1: The operators will improve more in respect to their execution time in the last three tasks compared to the first three.**
  They will get better at estimating depth and using their HUD in controlling Robothespian at extreme positions.

- **H2: The operators will accidentally knock down the object less in the last three tasks compared to the first three.**
  They will get better acclimated to the handling of Robothespian.

4.3.3 Stage 3: Shape Matching

In this stage, the participant performs the aforementioned second task six times in a row.

4.3.3.1 Task

The participant is given the object with the rectangular base by the experimenter and has to place it inside a box with a hole on its lid that matches the base. Since for these positions the robot is unable to place it gently inside, the operator is expected to drop it from a certain height after having estimated the angle where its base and the hole match. If the object lands outside the box, the experimenter will give it back to the operator. The six positions are chosen based on the extreme limits of Robothespian's arm as shown in Figure 4.2, like before.

4.3.3.2 Metrics and Measurement

For each trial, a number of metrics are recorded. These metrics are the time from the first movement of Robothespian's arm to the point where the object is placed inside the box and the number of times the object had to be reset due to the participant missing the hole.

4.3.3.3 Expected Outcome

We have two hypotheses regarding this stage:

- **H3: The operators will improve more in respect to their execution time in the last three tasks compared to the first three.**
  They will get better at estimating depth and using their HUD in controlling Robothespian at extreme positions.

- **H4: The operators will make more mistakes matching the shapes in the first three tasks compared to the last three.**
  They will get better acclimated depth perception and handling of Robothespian.

4.3.4 Stage 4: Conversation

After getting feedback from the pilot study participants, this stage was added. It serves as an enhancement of the illusion of body embodiment in which the operator is participating in a social interaction with the experimenter.

4.3.4.1 Task

The participant is instructed to have a conversation with the experimenter. The experimenter starts talking about random everyday stuff that might be of interest, and the operator is encouraged to take the reins of the conversation gradually. On the last few remaining minutes of the social interaction, the experimenter will sit on the right of the robot, making it known to the operator and setting up the next task.

4.3.4.2 Metrics and Measurement

There are no metrics captured in this stage. Its sole function is to give the participant time to get acclimated to using Robothespian in an everyday scenario, like that of conversing with another human.

4.3.4.3 Expected Outcome

The aim of this stage is to make the participant feel more comfortable with his new robotic body. It was mentioned by the users of the pilot study that only executing tasks that involved facing down to the tools felt non realistic and it hindered their feeling of embodiment. While they felt present at the robot's location, they could not feel as embodied as expected. However, the involvement of social interaction is expected to encourage the participants to accept this robotic body more as their own and use body language while conversing unobstructed.

4.3.5 Stage 5: Instinctive Reaction

In this stage, the participant is issued commands by the experimenter in fast succession and is asked to execute them as fast as possible.

4.3.5.1 Task

The participant is instructed to execute a series of commands issued by the experimenter. Those commands are fast and are randomly selected and repeated from a list. These commands are: ”Look at your left hand”, ”Look at your right hand”, ”Look at the table”, ”Look at the box”, Look at the ceiling” and the final command is always ”Look at me”.

4.3.5.2 Metrics and Measurement

The metrics that are captured at this stage are the direction the operator looked, as well as the explanation that he gives to the experimenter on how he tried to find him.

4.3.5.3 Expected Outcome

The aim of this stage is to strip the participant of time to think on their decisions and have them act on instinct. If the participant feels the illusion of body ownership transfer to Robothespian, he is going to look at the experimenter based on the robot's position, so he will have to turn his head to the right, as seen on Figure 4.3. The robot is stationed at the upper left corner of the room, the experimenter at the upper right and the operator is faced towards the right wall and is located on the lower right corner of the room.

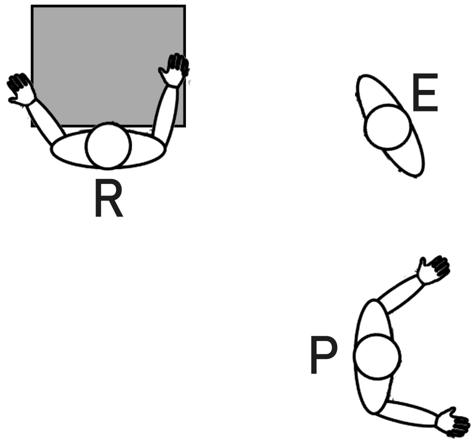

Figure 4.3: The positions of the Robot, Experimenter and Participant during the exam.

We have one hypothesis regarding this stage:

- **H5: The participant will try to locate the experimenter using the robot's position as point of origin.**

  They will turn their head to the right, as if they were in Robothespian's position.

Chapter 5

Protocol

5.1 Execution

Two studies were conducted for this dissertation, a smaller pilot one and a larger final one. The pilot study was conducted over one day with four participants who were recruited from undergraduate and postgraduate students from the UCL's Faculty of Engineering and were offered £5 as compensation for their participation in the study. The final case study was conducted over four days with twenty participants who were recruited from an external website and were offered £10 as compensation for their participation in the study.

The feedback procured from the participants of the pilot study served as a basis for the revision of the original protocol (see Appendix G) and inclusion of the Stage 4 of the experiment and the refinement of the final protocol (see Appendix H). No participant in either studies experienced any symptoms that hindered their performance or caused them to withdraw from the study.

A live performance by the supervising professor can be found in Youtube, titled "Robothespian Teleoperation in Virtual Reality at UCL", posted by the user Nikolaos Tsikinis.

5.2 Protocol

This section describes the revised protocol that was used to collect the data on the large final user study that led to the results discussed in this report.

5.2.1 Introduction, Demonstration and Tutorial

First, the participant is introduced to the experimenter and they are provided with the participant information sheet and the participant consent form (see appendices K and L). They are given time to read and sign both documents. After that, they are introduced to the hardware they are going to use and are shown the limits of Robothespian's movements live. Finally, an explanation is given of the controls and of the information that will be displayed on the HUD.

5.2.2 Calibration and Training Stage

After the completion of the introduction, the participant is shown where to stand and is given a brief explanation of the VR headset and controllers. After placing the HMD on the participant's head and adjusting it, the participant is instructed to extend their arms into the T-Pose and the experimenter presses the hotkey that calibrates the Bodyguard avatar based on the size of the participant. Then the experimenter reads the instructions for the training stages described in section 4.3.1 to the participant. After the five minutes pass or the participant feels ready, the experimenter presses the hotkey that enables the control of Robothespian. The experimenter is following the procedure that is explained in the aforementioned section.

5.2.3 Grabbing Stage

After the participant finishes the training stage, the experimenter introduces them to the next task. The experimenter explains the grabbing task, described in section 4.3.2 with the help of a demonstration of the task in real time in front of the robot. They are then informed that the task will be performed six times. Any questions the participant has at this point are answered by the experimenter. When ready, the participant performs the task. The experimenter then places the object in its next set position. When the object is re-positioned, the participant is instructed to perform the task again.

5.2.4 Shape Matching Stage

After the participant finishes the grabbing stage, the experimenter introduces them to the next task. The experimenter explains the matching task, described in section 4.3.3 with the help of a demonstration of the task in real time in front of the robot. They are then informed that the task will be performed six times. Any questions the participant has at this point are answered by the experimenters. When ready, the participant performs the task. The experimenter then places the box in its next set position and places the object in the robot's hand. When that is done, the participant is instructed to perform the task again.

5.2.5 Conversation Stage

After the participant finishes the shape matching stage, the experimenter introduces them to the next task. The experimenter explains the conversation task, described in section 4.3.4. The experimenter moves in front of the robot in a normal conversation distance and starts talking about general topics, trying to find something that engages the participant. After a suitable topic is found, the experimenter attempts to encourage the participant into taking the reins of the conversation, making him feel more comfortable, all while noting his use of body language. During the last minutes, the experimenter moves to the right of the robot, entering and exiting his vision range, setting the participant up for the next stage.

5.2.6 Instinctive Reaction Stage

After the participant finishes the shape conversation stage, the experimenter introduces them to the next task. The experimenter explains the reaction task, described in section 4.3.5. The experimenter starts issuing orders in fast succession and multiple times, and when they judge the participant is acting on instinct, they issue the final command for the participant to locate them. The experimenter is taking note of their reactions during that time.

5.2.7 Debriefing

Finally, the participant is helped out of the headset by the experimenter and asked if they have any further comments on their experience that they want to share. If they do, the experimenters makes note of them. Then the participant is given a questionnaire in which they are asked to give information about their experience during the experiments as well as their background, like age, gender and previous experience with virtual reality. This questionnaire also changed based on feedback from the pilot study, and the original can be seen at Appendix I), while the final one at Appendix J. The participant is then given their compensation and is asked to confirm this by signature.

Chapter 6

Results

The findings of the two conducted user studies are going to be presented in this chapter, beginning with the pilot study. The presentation of the results is identical in both studies and organized as follows: First, general information are given about the demographics of the participants. Afterwards, the visualization and analysis of the data is split into two categories, one outlining the agency and motor performance of the participants, which includes the ”Grabbing Objects” and ”Shape Matching” stages, and the other the degree of body ownership transfer achievement, which includes the ”Instinctive Reaction” stage. In each category, an explanation is given on how the data is prepared and visualized and on how it is interpreted. Finally, a combined analysis is provided outlining the degree that the participants experienced the phenomenon of embodiment.

6.1 Pilot Study

While the pilot study had a very small number of participants, certain trends were easy to discern. Four people participated and their demographics can be seen in Figure 6.10.

6.1.1 Agency and Motor Performance

6.1.1.1 Data Preparation and Visualization

Each participant's task execution time and number of resets needed can be seen in Table 6.1 for the ”Grabbing Objects” task and in Table 6.2 for the ”Shape

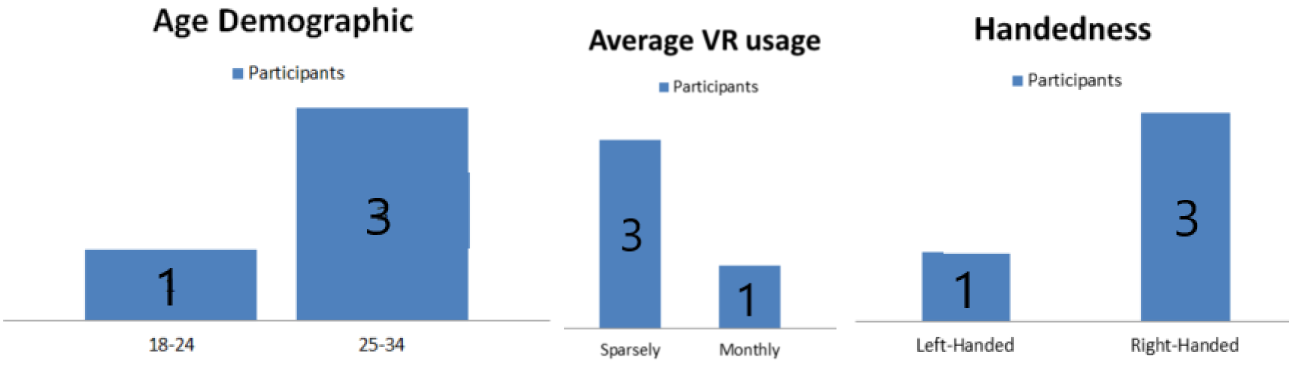

Figure 6.1: Participant demographic statistics in pilot study.

Matching" task.

Participant	Task 1	Task 2	Task 3	Task 4	Task 5	Task 6
1	25 (0)	37 (1)	36 (1)	30 (0)	83 (1)	59 (1)
2	20 (0)	87 (2)	14 (1)	38 (1)	12 (0)	80 (1)
3	14 (0)	16 (0)	18 (0)	20 (0)	16 (0)	76 (2)
4	45 (0)	42 (0)	28 (0)	15 (0)	12 (0)	110 (4)

Table 6.1: Pilot study participant performance results in Grabbing Objects. Times are in seconds. The numbers in parentheses are the number of object resets.

Participant	Task 1	Task 2	Task 3	Task 4	Task 5	Task 6
1	14 (0)	37 (1)	48 (0)	22 (1)	14 (0)	25 (1)
2	20 (1)	40 (1)	23 (1)	202 (6)	27 (0)	40 (1)
3	24 (1)	7 (0)	47 (3)	8 (0)	14 (0)	15 (0)
4	24 (1)	14 (1)	25 (1)	17 (1)	5 (0)	10 (0)

Table 6.2: Pilot study participant performance results in Matching Shapes. Times are in seconds. The numbers in parentheses are the number of object resets.

To identify the change in performance over the six grabbing tasks, the linear regression was calculated for every metric for each participant. In particular, to measure the change in a participant's execution time, the difference between each set set of tasks is measured, as outlined in sections 4.3.2.3 and 4.3.3.3. For example, in the first stage Participant 1 performed task 2 twelve seconds slower than task 1, and task 3 one second faster than task 2. This trend is repeated in the last set of tasks, where he performs task 5 slower than task 4, but task 6 is executed faster than task 5. The resulting graphs can be found in the Appendix C, and the example of

Participant 1 can be seen in 6.2, where we can see his time graph and the trend-lines for the two groups of tasks. Solid lines symbolize members of the first three tasks, while dotted lines stand for tasks 4, 5 and 6. To compare the changes in performance, these regressions were brought to a common starting point, by setting their constant to zero. Note that a steeper descent stands for a positive improvement, as in all the metrics a lower value represents a better performance. So, Participant 1 showed degradation in the performance of the last three tasks compared to the first three, while requiring the same number of object resets.

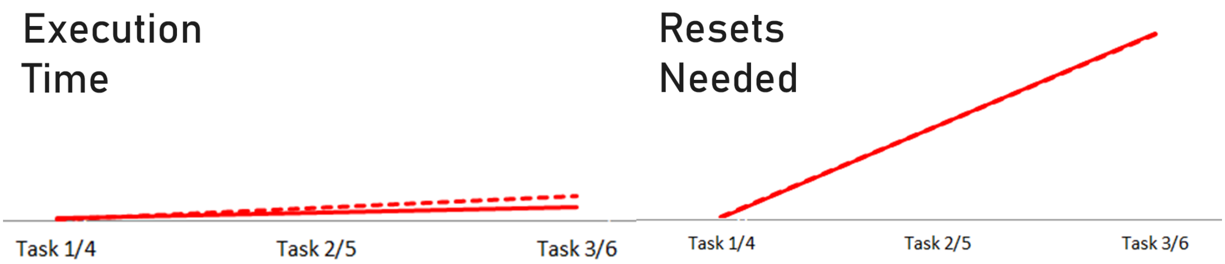

Figure 6.2: Pilot study performance trends of Participant 1 in "Grabbing Objects" task with respect to execution time and object reset times.

The resulting regressions regarding both the execution time and number of resets needed for all participants are displayed in Figure 6.3 for the "Grabbing Objects" task and in Figure 6.4 for the "Shape Matching" task.

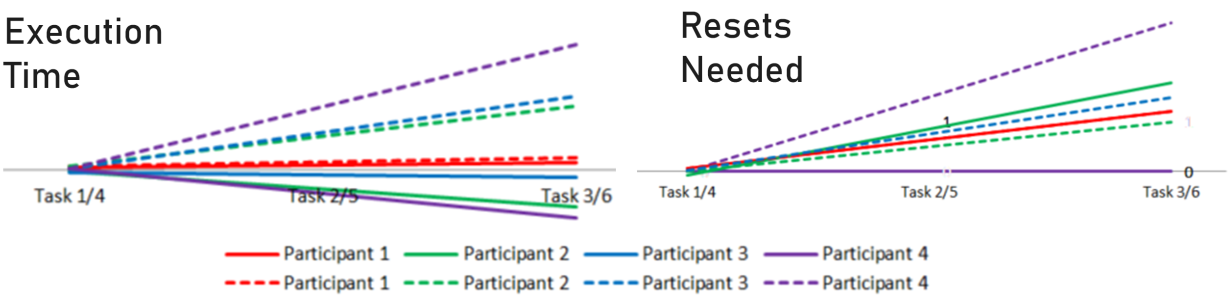

Figure 6.3: Pilot study performance trends of participants with respect to execution time and object reset times in "Grabbing Objects" task.

To determine the overall trend, the mean of the coefficients by group was calculated, leading to one regression representing the average performance change for tasks 1,2 and 3 and one - dotted - regression representing the average performance change for the rest. This graph is displayed in Figure 6.5 for the "Grabbing Objects"

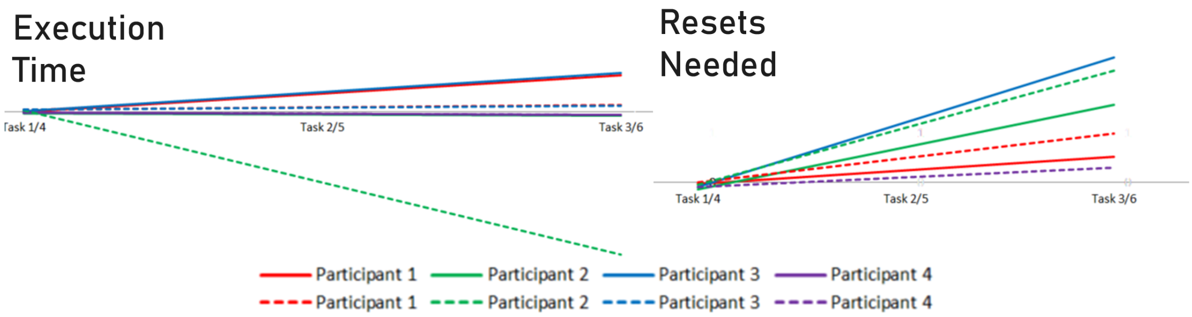

Figure 6.4: Pilot study performance trends of participants with respect to execution time and object reset times in ”Shape Matching” task.

task and in Figure 6.6 for the ”Shape Matching” task.

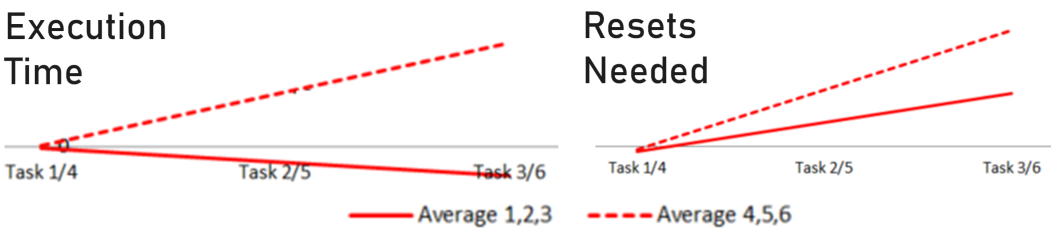

Figure 6.5: Pilot study average performance trends change in ”Grabbing Objects” task.

Figure 6.6: Pilot study average performance trends change in ”Shape Matching” task.

Finally, the participants’ feedback to the relevant statements are shown in Figure 6.7. Those statements were:

- Q4: It felt like I could control the virtual body as if it was my own body.
- Q5: The movements of the virtual body were caused by my movements.
- Q6: I felt as if the movements of the virtual body were influencing my own movements.
- Q7: I felt as if the virtual body was moving by itself.

- Q10: During calibration I felt as if my real body were drifting towards the virtual body or as if the virtual body were drifting towards my real body.
- Q11: If Right-Handed, did you feel like you had difficulty in controlling the Left arm?
- Q12: I was able to adjust to the virtual avatar's controls easily.
- Q13: I was able to adjust to the robot's controls easily.
- Q14: I was able to adjust my depth perception when operating the robot easily.
- Q15: I found the HUD interface helpful in controlling the robot.

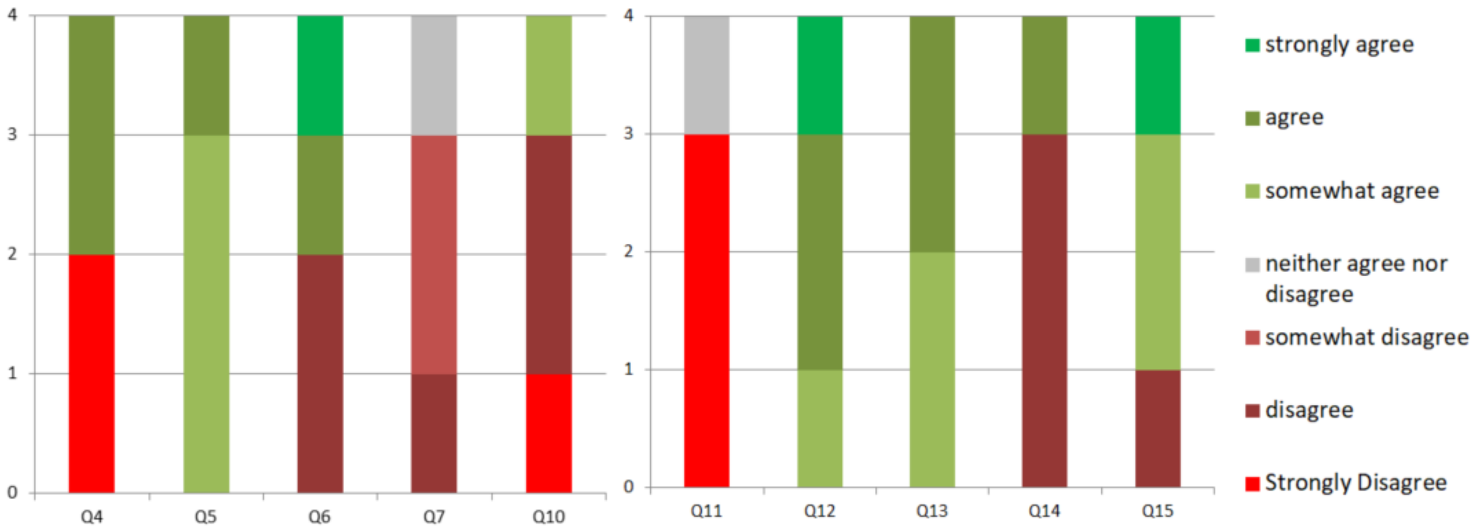

Figure 6.7: Pilot study agency and motor performance answers to Questionnaire.

6.1.1.2 Interpretation

In section 4.3.2.3, we made two hypotheses regarding the "Grabbing Objects" stage:

- **H1: The operators will improve more in respect to their execution time in the last three tasks compared to the first three.**
  From observing the regressions from all participants in Figure 6.3 and the averaged performance changes in Figure 6.5, it is obvious that they improved

more overall during the first sets of the tasks. As was mentioned, a steeper descent stands for a positive improvement, as in all the metrics a lower value represents a better performance. In this case, the averaged metrics exhibit a steep ascent for the second set of tasks and a slight descent for the first set of tasks, indicating that the participants improved their performance more in these. Even after further examining each participant's performance results, no evidence can be found that contradict this, so the hypothesis is not supported.

- **H2: The operators will accidentally knock down the object less in the last three tasks compared to the first three.**
  By observing the aforementioned averaged performance changes, it can be seen that the participants required more resets of the object in the second set of tasks. However, when examining them individually in the regressions from all participants, this is not immediately obvious, as the regressions seem to be balanced. This balance is disturbed by Participant 4, who needed way more resets than the rest, and skews the average against the second tasks. Still, it hard to support this hypothesis judging from the presentation of the averages.

In section 4.3.3.3, we made two hypotheses regarding the "Shape Matching" stage:

- **H3: The operators will improve more in respect to their execution time in the last three tasks compared to the first three.**
  From observing the averaged performance changes in Figure 6.6, it is obvious that they improved more overall in the second sets of the tasks. The averaged metrics exhibit a steep descent for the second set of tasks and a slight ascent for the first set of tasks, indicating that the participants actually worsened their performance for it. However, this is not clear during the examination of the regressions from all participants in Figure 6.4. Even though the improvement of Participant 2 is so great that it skewers the averaged regression in favor of the hypothesis, it can also be seen that more participants performed better in the second set of tasks.

- **H4: The operators will make more mistakes matching the shapes in the first three tasks compared to the last three.**

  By observing the aforementioned averaged performance changes, it is clear that while the participants did not positively improve their performance in either set of tasks, they did not require as many resets in the first set as much as in the second set. Upon closer examination of their results and the aforementioned regressions from all participants, it can be seen that the difference is borderline. Still, the hypothesis cannot be clearly supported, as the averaged regression supports it without the existence of any major outliers.

The answers of the Questionnaire's questions regarding the agency and control of Robothespian seem to indicate that the participants felt confident they could control the robot adequately. They felt that the movements of the robotic body were caused by their movements, without observing movements that were not performed by them. They were able to adjust to the controls easily and the right-handed participants did not encounter any difficulty in performing the tasks using their left hand. However, they did not feel like they adjusted easily to the lacking depth perception the camera offers, even though they performed better in the tasks of the second stage, which required adequate sense of depth to match the objects' shapes. Finally, only half of them felt they could control the robot as if it were their own body, mainly due to the imitated movement capabilities of Robothespian and their inability to use their right hand for the tasks.

6.1.1.3 Conclusion

We failed to support the hypothesis that the operators will display improvement in their execution time in tasks four to six compared to that of in tasks one to three in the ”Grabbing Objects” stage. We failed to support the hypothesis that the operators will accidentally knock down the object less in positions four to six than in one to three in the ”Grabbing Objects” stage. We managed to support the hypothesis that the operators will display improvement in their execution time in tasks four to six compared to that of in tasks one to three in the ”Shape Matching” stage. We managed to support the hypothesis that the operators will accidentally knock down the object less in positions four to six than in one to three in the "Shape Matching" stage. According to the feedback from the Questionnaire, the participants felt they could control the robot to a satisfactory degree.

6.1.2 Embodiment and Instinctive Response

6.1.2.1 Data Preparation and Visualization

The reaction of the participants can be seen in Figure 6.8. Only the immediate reaction after issuing the order to look at the experimenter is noted. The direction that is presented here is based from the perspective of their physical body.

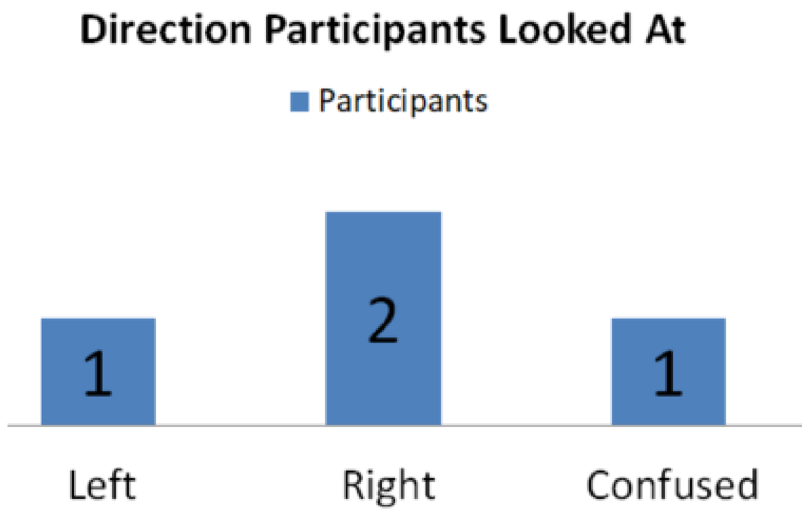

Figure 6.8: Direction participants looked at in pilot study's "Instinctive Reaction" task. Perspective is based on their physical body.

Additionally, the participants' feedback to the relevant statements are shown in Figure 6.9. Those statements were:

- Q1: I felt as if the virtual body I saw when I looked down was my body.
- Q2: It felt as if the virtual body I saw was someone else.
- Q3: It seemed as if I might have more than one body.
- Q8: I felt as if my body was located where I saw the virtual body.
- Q9: I felt out of my body.

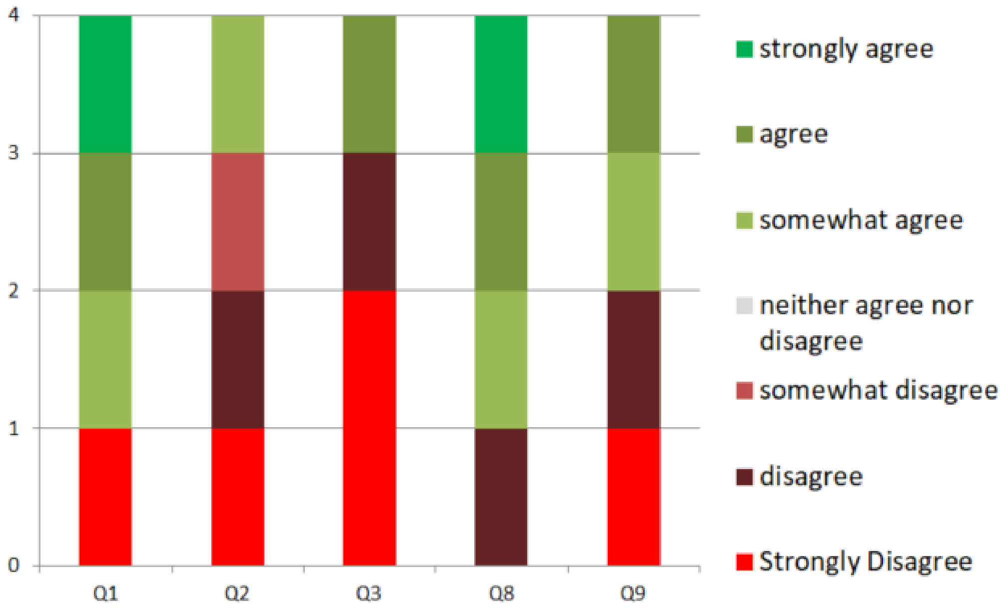

Figure 6.9: Pilot study embodiment answers to Questionnaire.

6.1.2.2 Interpretation

In section 4.3.5.3, we made one hypothesis regarding the "Instinctive Reaction" stage:

- **H5: The participant will try to locate the experimenter using the robot's position as point of origin.**

  From observing the participants' reactions it can be discerned that half of them turned right, which is a reaction expected from someone that reacted based on Robothespian's position. For the other half, one half of it turned left, a reaction expected from someone that bases his reactions on his physical body, while the other half was confused and turned all over the place, meaning they had conflicts on how to execute the order.

The reactions to the Questionnaire's statements regarding the body ownership transfer to Robothespian seem to indicate that the participants experienced the phenomenon of embodiment and telepresence at a considerable degree. Three quarters of them felt that they owned the robot's body when they looked at it and believed they only had one body in total. Finally, most of them believed they were located at the robot's place, something that is confirmed by their reactions.

6.1.2.3 Conclusion

We managed to somewhat support the hypothesis regarding how the participants would try to locate the experimenter, based on the small number of users.

6.1.3 Pilot Study Conclusion

Overall, after reviewing the participants' comments and reactions to the statements in the questionnaire, it can be stated that the control system proposed by this dissertation was successful in creating the illusion of body ownership transfer and telepresence in a humanoid robot. The participants were able to control it adequately, even being handicapped in the sense that three out of four were not using their dominant hand in performing the tasks. Additionally, while there might be many methods in trying to locate the experimenter, like focusing on their voice while issuing orders or having a mental map of the room layout and acting based on that, all of them are differently oriented based on their point of origin. No matter how they try to locate the experimenter, they would turn right if they were feeling to be positioned in the robot's place and left if they acted based on their real body's perception. An interesting thing to notice is that only one out of the four participants felt the mechanical look of the robot forbidding in allowing themselves to feel embodied in it. That, however, did not prohibit them from operating the robot efficiently.

6.2 User Study

Due to the small number of participants that showed interest in the pilot study, its results are not be considered representative and further experimentation was required. Twenty people participated in the second, larger user study and their demographics can be seen in Figure 6.10.

6.2.1 Agency and Motor Performance

6.2.1.1 Data Preparation and Visualization

Each participant's task execution time and number of resets needed can be seen in Table 6.3 for the ”Grabbing Objects” task and in Table 6.4 for the ”Shape

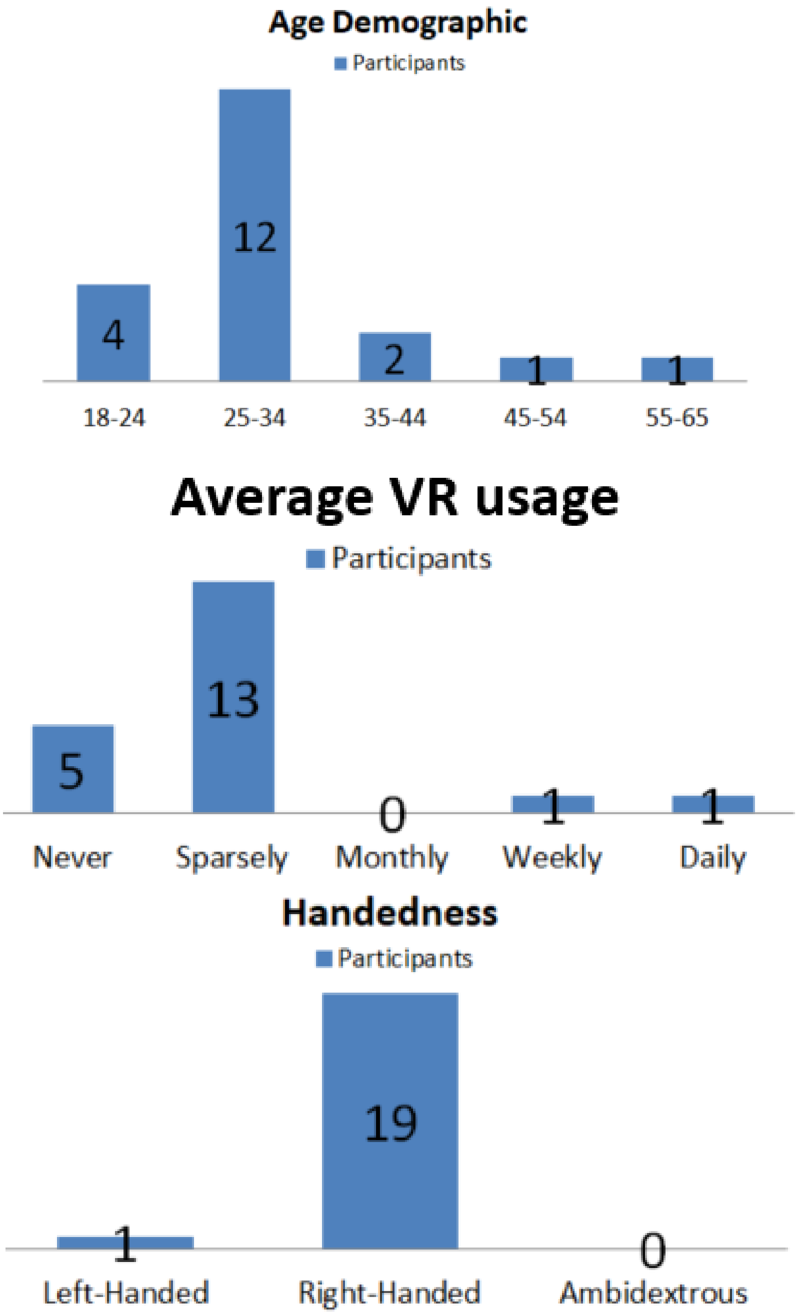

Figure 6.10: Participant demographic statistics in user study.

Matching" task.

Participant	Task 1	Task 2	Task 3	Task 4	Task 5	Task 6
1	14 (0)	9 (0)	46 (2)	22 (0)	25 (0)	48 (1)
2	10 (0)	55 (6)	21 (0)	19 (0)	7 (0)	15 (0)
3	28 (0)	36 (1)	7 (0)	10 (0)	6 (0)	10 (0)
4	9 (0)	13 (0)	29 (3)	6 (0)	4 (0)	7 (0)
5	7 (0)	19 (0)	15 (0)	10 (0)	5 (0)	6 (0)
6	14 (2)	16 (0)	10 (0)	19 (0)	5 (0)	8 (0)
7	9 (0)	6 (0)	9 (0)	7 (0)	5 (0)	8 (0)
8	20 (0)	17 (0)	42 (0)	30 (0)	11 (0)	11 (0)
9	16 (0)	13 (0)	9 (0)	17 (0)	11 (0)	12 (0)
10	6 (0)	9 (0)	12 (0)	15 (0)	19 (0)	27 (0)
11	8 (0)	13 (0)	18 (0)	12 (0)	45 (0)	21 (1)
12	29 (0)	10 (0)	11 (0)	67 (0)	20 (0)	24 (0)
13	12 (0)	14 (0)	26 (0)	33 (2)	10 (0)	20 (0)
14	22 (0)	26 (0)	25 (0)	21 (0)	22 (0)	18 (0)
15	10 (0)	25 (0)	13 (0)	10 (0)	11 (0)	8 (0)
16	54 (0)	37 (0)	16 (0)	79 (2)	24 (1)	24 (0)
17	17 (1)	14 (0)	29 (1)	6 (0)	9 (1)	6 (0)
18	20 (0)	27 (0)	33 (0)	19 (0)	12 (0)	9 (0)
19	10 (0)	7 (0)	4 (0)	11 (0)	17 (0)	37 (2)
20	14 (0)	7 (0)	6 (0)	21 (1)	15 (0)	26 (1)

Table 6.3: User study participant performance results in Grabbing Objects. Times are in seconds. The numbers in parentheses are the number of object resets.

Participant	Task 1	Task 2	Task 3	Task 4	Task 5	Task 6
1	45 (0)	22 (0)	23 (0)	13 (0)	10 (0)	9 (0)
2	20 (0)	27 (1)	26 (1)	8 (0)	22 (1)	20 (0)
3	24 (0)	22 (0)	47 (1)	17 (0)	14 (0)	15 (0)
4	14 (1)	7 (0)	26 (2)	9 (0)	6 (0)	6 (0)
5	12 (0)	10 (0)	4 (0)	6 (0)	3 (0)	4 (0)
6	42 (0)	13 (0)	5 (0)	10 (0)	10 (0)	8 (0)
7	8 (0)	6 (0)	4 (0)	5 (0)	3 (0)	3 (0)
8	47 (0)	64 (0)	46 (0)	48 (0)	33 (0)	49 (0)
9	13 (0)	27 (1)	15 (0)	26 (0)	10 (0)	4 (0)
10	19 (0)	7 (0)	10 (0)	9 (0)	9 (0)	6 (0)
11	26 (1)	40 (2)	14 (0)	7 (0)	4 (0)	8 (0)
12	19 (0)	17 (0)	7 (0)	12 (0)	5 (0)	14 (1)
13	14 (0)	14 (0)	13 (0)	48 (2)	4 (0)	3 (0)
14	25 (0)	32 (0)	18 (0)	23 (0)	27 (0)	31 (0)
15	11 (0)	9 (0)	14 (1)	18 (0)	23 (1)	26 (0)
16	15 (0)	7 (0)	13 (0)	5 (0)	4 (0)	6 (0)
17	11 (0)	5 (0)	11 (0)	8 (0)	15 (0)	6 (0)
18	26 (0)	35 (0)	46 (0)	27 (0)	23 (0)	15 (0)
19	6 (0)	10 (1)	7 (0)	11 (0)	4 (0)	8 (0)
20	6 (0)	5 (0)	6 (0)	7 (0)	5 (0)	5 (0)

Table 6.4: User study participant performance results in Matching Shapes. Times are in seconds. The numbers in parentheses are the number of object resets.

Like before the linear regression was calculated for every metric for each participant. In particular, to measure the change in a participant's execution time, the difference between each set set of tasks is measured, as outlined in sections 4.3.2.3 and 4.3.3.3. The resulting graphs can be found in the Appendix D. Solid lines symbolize members of the first three tasks, while dotted lines stand for tasks 4, 5 and 6. To compare the changes in performance, these regressions were brought to a common starting point, by setting their constant to zero. The resulting regressions regarding both the execution time and number of resets needed for all participants are displayed in Figure 6.11 for the "Grabbing Objects" task and in Figure 6.12 for the "Shape Matching" task.

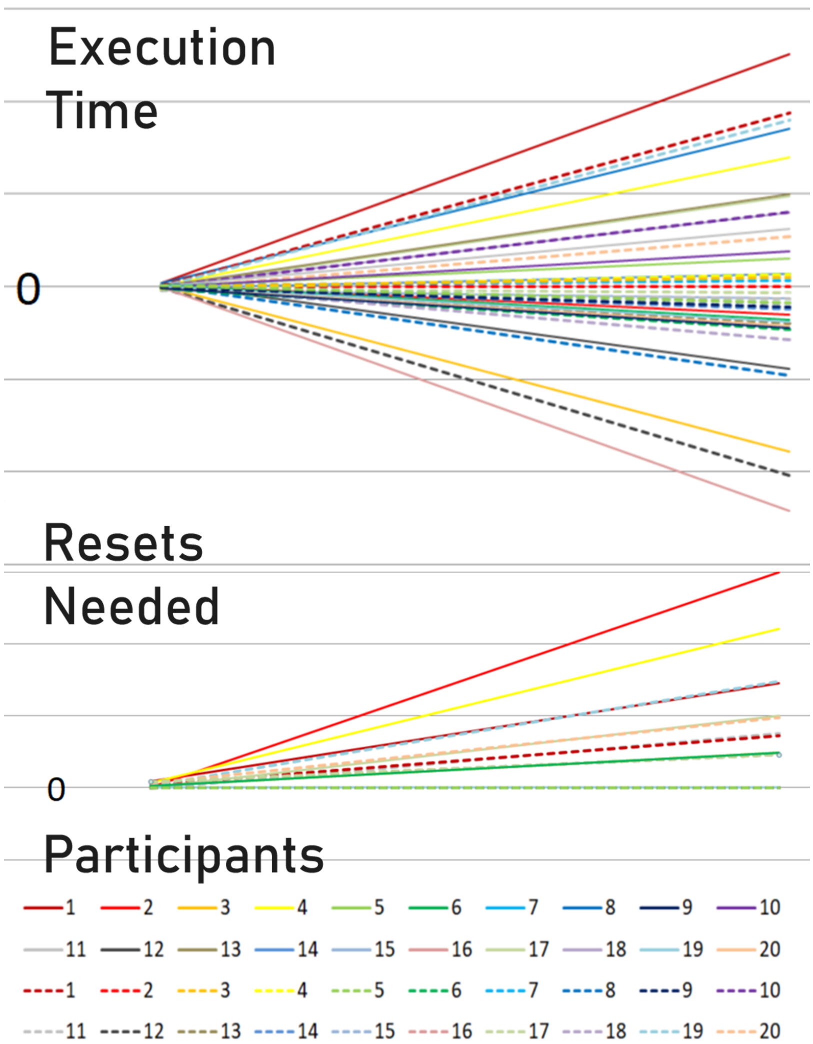

Figure 6.11: User study performance trends of participants with respect to execution time and object reset times in "Grabbing Objects" task.

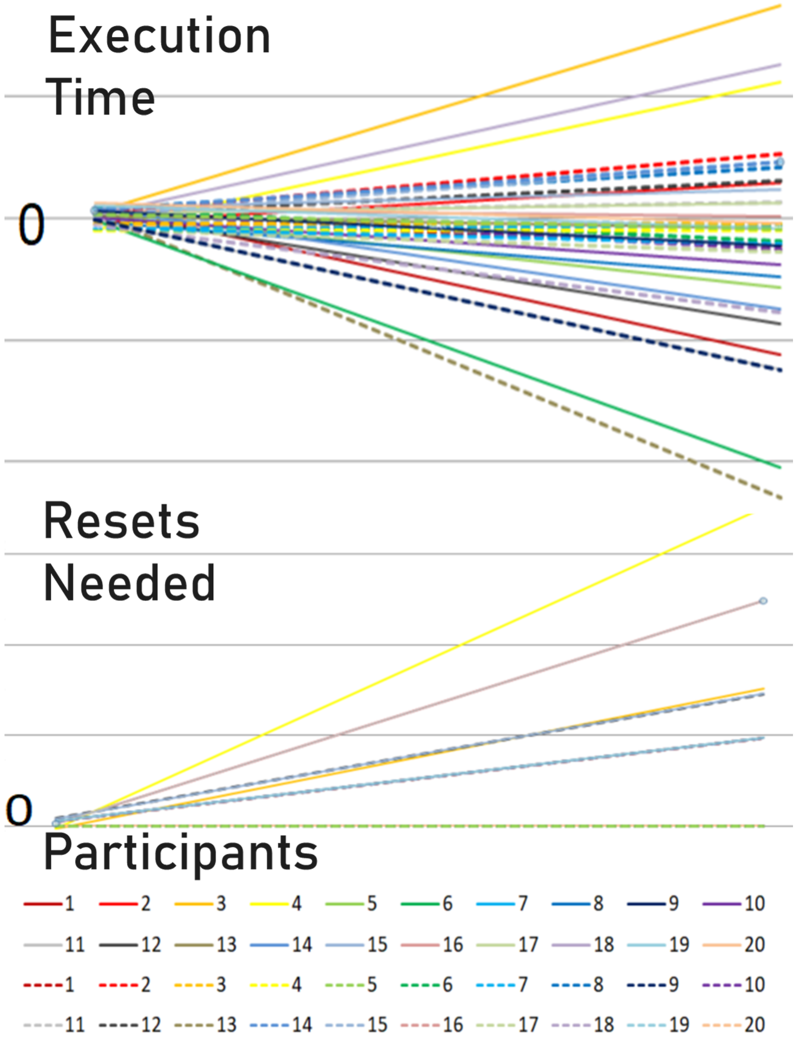

Figure 6.12: User study performance trends of participants with respect to execution time and object reset times in ”Shape Matching” task.

To determine the overall trend, the mean average of the coefficients by group was calculated, leading to one regression representing the average performance change for tasks 1,2 and 3 and one - dotted - regression representing the average performance change for the rest. This graph is displayed in Figure 6.13 for the ”Grabbing Objects” task and in Figure 6.14 for the ”Shape Matching” task.

Finally, the participants’ feedback to the relevant statements are shown in Figure 6.15. Those statements were:

- Q2: It felt like I could control the virtual body as if it was my own body.
- Q3: The movements of the robotic body were caused by my movements.

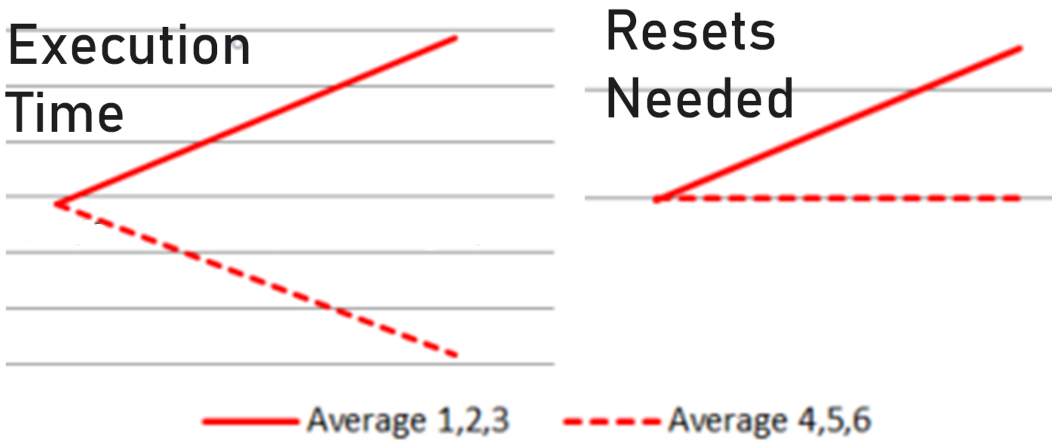

Figure 6.13: User study average performance trends change in "Grabbing Objects" task.

Figure 6.14: User study average performance trends change in "Shape Matching" task.

- Q4: I felt as if the robotic body was moving by itself.
- Q6: I felt as if the robotic body was matching the movements of my real body.
- Q7: I was able to adjust to the controls easily.
- Q8: I found the HUD interface useful in controlling the robot.
- Q9: Even though the depth perception of the camera was limited, I feel like I sufficiently adjusted to it.
- Q11: Did you find it easy to perform the task left handed?

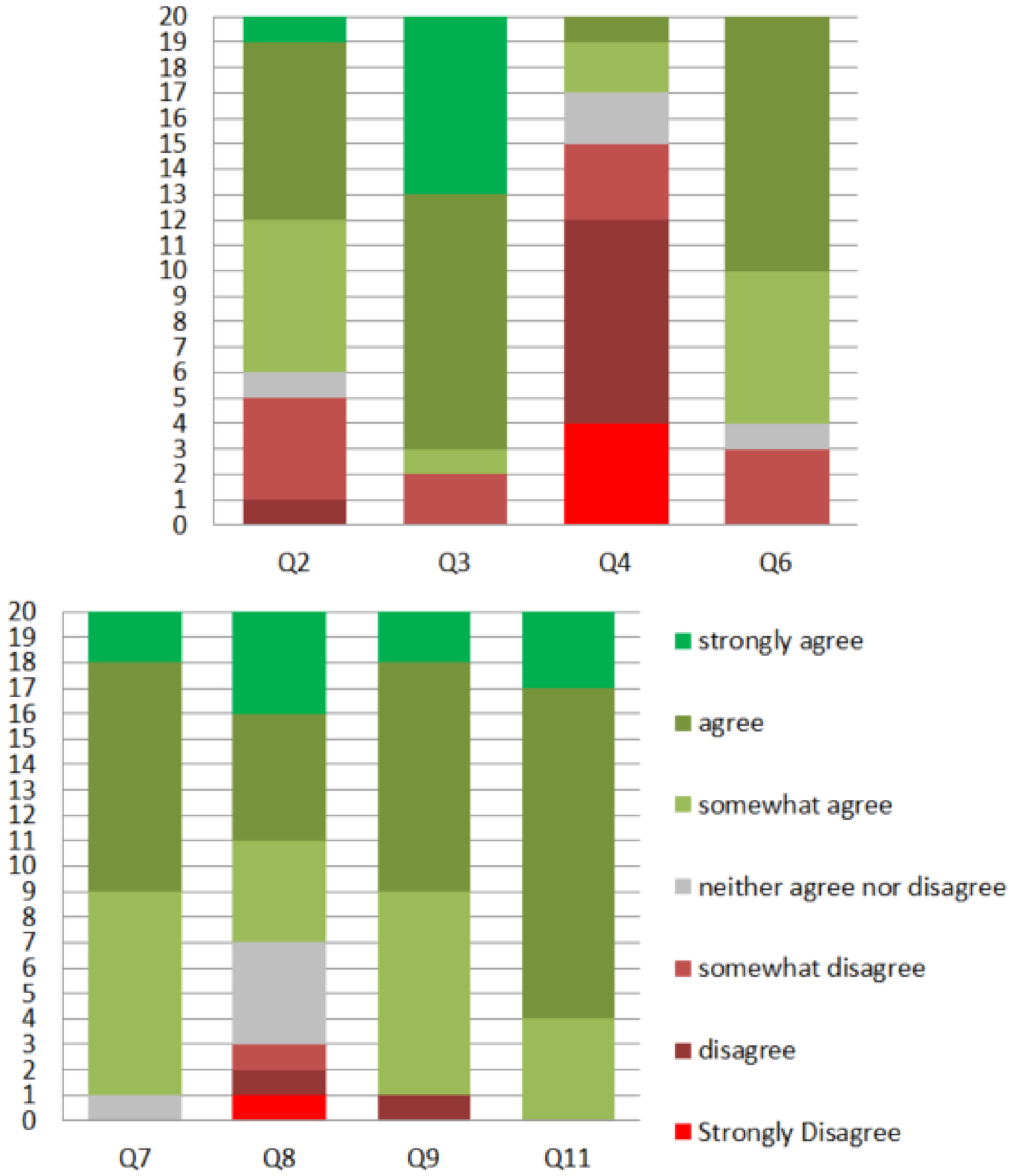

Figure 6.15: User study agency and motor performance answers to Questionnaire.

6.2.1.2 Interpretation

In section 4.3.2.3, we made two hypotheses regarding the ”Grabbing Objects” stage:

- **H1: The operators will improve more in respect to their execution time in the last three tasks compared to the first three.**

  By examining regressions from all participants in Figure 6.11 we can not immediately reach a conclusion, due to the number of lines. But, we can see that they are evenly spread and there do not appear to be any extreme outliers that might skew the outcome. The averaged performance changes in Figure 6.13, however, provide an interesting outlook on the participant's performance. It seems that their performance improved at the second set of tasks at the same rate that it diminished at the first. This supports this hypothesis and it would be interesting to examine whether this happened by chance in future research.

- **H2: The operators will accidentally knock down the object less in the last three tasks compared to the first three.**
  By observing the aforementioned averaged performance changes, it can be seen that the participants required more resets of the object in the first set of tasks, while they required a steady average number of resets on the second set. That can be further surmised from examining the results, where is it seen that most participants did not required any reset at the second set of tasks, supporting this hypothesis since possibly they got adequately adept at controlling Robothespian.

In section 4.3.3.3, we made two hypotheses regarding the "Shape Matching" stage:

- **H3: The operators will improve more in respect to their execution time in the last three tasks compared to the first three.**
  From observing the averaged performance changes in Figure 6.14 it can be seen that the participants displayed improvement at both sets of tasks, with the second set having a slight advantage. But, this is not clear upon examining the regressions from all participants in Figure 6.12, due to the number of lines. But, we can see that they are, again, evenly spread and there do not appear to be any extreme outliers that might skew the outcome. The overall improvement in their performance supports this hypothesis and, compared to the previous experiment stage, can be attributed to their getting used to the controls and depth perception.

- **H4: The operators will make more mistakes matching the shapes in the**

first three tasks compared to the last three.

By observing the aforementioned averaged performance changes and the raw result data, it is clear that while the participants improved their performance in both stages - as outlined previously -, they made less mistakes in the second stage, thus supporting this hypothesis.

The answers of the Questionnaire's questions regarding the agency and control of Robothespian seem to indicate that the vast majority of the participants felt they were in total control of the robot. They felt that the its movements were caused by them, without observing movements that were not performed by them. They were able to adjust to the controls easily and none encountered any difficulty in performing the tasks using their left hand. Additionally, 95% felt like they adjusted easily to the lacking depth perception the camera offers, in contrast to the pilot study. Finally, 85% of them felt that the HUD was useful to them.

6.2.1.3 Conclusion

We managed to support the hypothesis that the operators will display improvement in their execution time in tasks four to six compared to that of in tasks one to three in the "Grabbing Objects" stage. We managed to support the hypothesis that the operators will accidentally knock down the object less in positions four to six than in one to three in the "Grabbing Objects" stage. We managed to confirm that the operators will display improvement in their execution time in tasks four to six compared to that of in tasks one to three in the "Shape Matching" stage. We managed to confirm that the operators will accidentally knock down the object less in positions four to six than in one to three in the "Shape Matching" stage. According to the feedback from the Questionnaire, the participants felt they could control the robot to a satisfactory degree.

6.2.2 Embodiment and Instinctive Response

6.2.2.1 Data Preparation and Visualization

The reaction of the participants can be seen in Figure 6.16. Only the immediate reaction after issuing the order to look at the experimenter is noted. Perpsective is based on the participant's physical body.

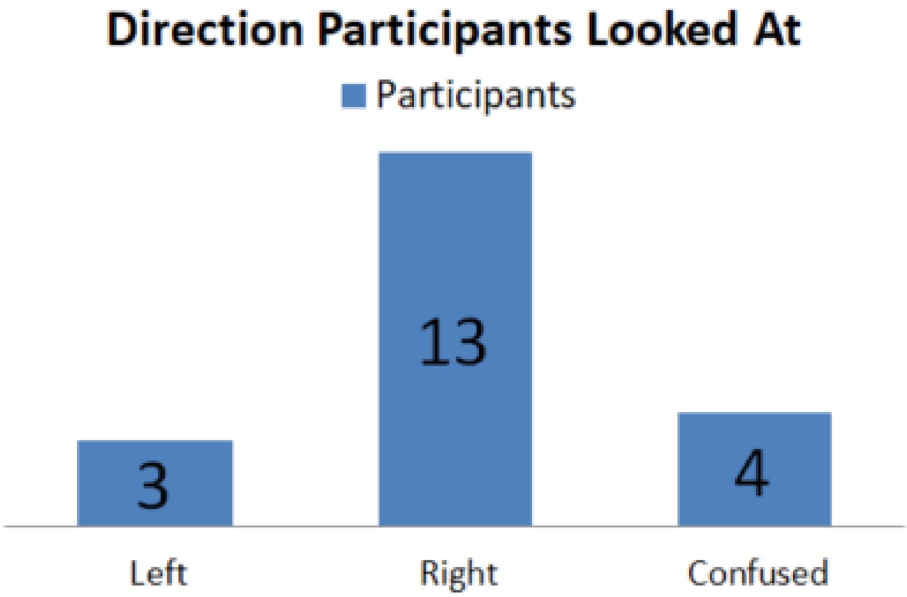

Figure 6.16: Direction participants looked at in user study's "Instinctive Reaction" task. Perspective is based on their physical body.

Additionally, the participants' feedback to the relevant statements are shown in Figure 6.17. Those statements were:

- Q1: I felt as if the virtual body I saw when I looked down was my body.
- Q5: It felt as if the virtual body I saw was someone else.
- Q10: When asked to locate the Experimenter, I tried to find him as if I were in the robot's place.

6.2.2.2 Interpretation

In section 4.3.5.3, we made one hypothesis regarding the "Instinctive Reaction" stage:

- **H5: The participant will try to locate the experimenter using the robot's position as point of origin.**

  From observing the participants' reactions it can be discerned that more than half of them (65%) turned right, which is a reaction expected from someone

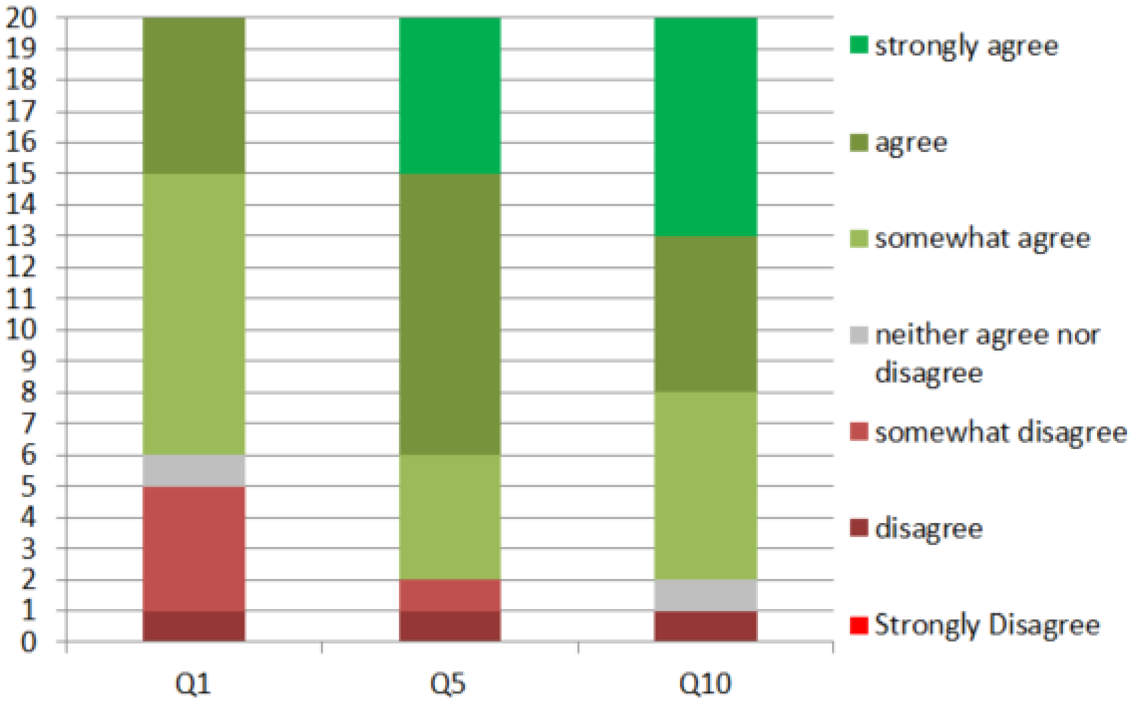

Figure 6.17: User study embodiment answers to questionnaire.

that reacted based on Robothespian's position. The minority (15%) turned left, a reaction expected from someone that bases his reactions on his physical body, while the rest (20%) were confused and either hesitated or turned all over the place, meaning they had conflicts on how to execute the order.

The reactions to the Questionnaire's statements regarding the body ownership transfer to Robothespian seem to indicate that the vast majority of the participants experienced the phenomenon of embodiment and telepresence at a considerable degree. Nearly three quarters (70%) of them felt that they owned the robot's body when they looked at it and the vast majority believed they only had one body in total. Finally, 90% of them tried to locate the experimenter as if they were the robot, meaning they felt embodied and present at its location.

6.2.2.3 Conclusion

We managed to support the hypothesis regarding how the participants would try to locate the experimenter.

6.2.3 User Study Conclusion

Overall, after reviewing the participants' comments and reactions to the statements in the questionnaire and after taking into account the results from the pilot study, it can be stated that the control system proposed by this dissertation was successful in creating the illusion of body ownership transfer and telepresence in a humanoid robot. The participants were able to control it adequately, even being handicapped in the sense that the vast majority was not using their dominant hand in performing the tasks. Additionally, while there might be many methods in trying to locate the experimenter, like focusing on their voice while issuing orders or having a mental map of the room layout and acting based on that, all of them are differently oriented based on their point of origin. No matter how they try to locate the experimenter, they would turn right if they were feeling to be positioned in the robot's place and left if they acted based on their real body's perception.

Chapter 7

Conclusions, Evaluation and Further Work

Over the course of the project, a VR teleoperation control system for the humanoid robot Robothespian was developed. This was then used to design experiments for a user study meant to evaluate the effectiveness of this system in creating the phenomena of body ownership transfer to the robot and telepresence to its position. The first study's feedback review resulted in changes of the protocol for the larger second one. Those changes proved successful in increasing the degree that the participant experienced these phenomena. Overall, the larger participant pool led to some clear trends, that allowed to support all the hypotheses. The teleoperation system that was developed utilizes any off-the-shelf VR headset and accessories and can facilitate the phenomena of telepresence and body ownership transfer to a remote robot, while allowing its operator to accurately control it.

Due to the time required for repairing Robothespian, it was decided that it would be impossible to have it fully operational in time for the user studies, since it displayed symptoms of breaking down shortly before the pilot study. That caused the exclusion of the tasks that utilized the right hand, as well as any torso movement. Due to the nature of this research, the skills of each participant were irrelevant, so there was no need to set VR or operational specific requirements, which helped greatly. Additionally, the visual feed was of sufficient quality and had very small delay, minimizing the chances of the participants experiencing any simulation sickness effect through prolonged exposure.

An obvious step in the future would be to repeat the study in a larger and revised form, with a Robothespian that is fully operational. A fully functional robot would allow for a wider range of tasks, which could test the operator's control capabilities further. Additionally, it might prove that being able to fully utilize both hands in any scenario, even when conversing, would further enhance the feeling of embodiment. An improved Robothespian model would also be more suitable for using a stereoscopic camera to provide the visual feedback to the operator, further improving their depth perception sense. Finally, if the participant pool is high enough, they can be split into two groups, one that has the social interaction stage and one that does not. This way, it would be possible to discern whether including social interactions plays any part in improving the phenomenon of body ownership transfer. A larger study would also be able to confirm if the tendencies observed in this report hold true for a larger population as well.

Acknowledgements

For the development of the application, the game engine Unity was used. The Unity packages used and their respective role in the creation of this project are listed in table 7.1. The rest of the work, as well as any adjustment mentioned, was done by the author of this report. I also would like to thank my supervisor Dr. David Swapp for his support.

Steam VR	[18]	Used to provide an API compatible with most retail VR equipment
Bodyguards	[19]	Used to create the virtual calibration avatar
Final IK	[22]	Used to provide the inverse kinematics solution for the virtual avatar

Table 7.1: List of external unity packages used and purpose.

Appendix A

System manual

A.1 Access to the Project Files

The project currently resides in private GitLab project, that can only be accessed by the author. An archived copy is also stored on the network drive of UCL's computer science department and can be obtained from Dr. David Swapp.

A.2 Requirements and Setup

The software was developed with the version 2018.3.3f1 of Unity. No further compilation is required. The Ricoh Theta V firmware version that was used was 3.00.1 released on May. 22, 2019. The computer that runs the Unity program has to have Ricoh Theta V's drivers installed. For this research version 3.10.1, released on May. 17, 2019, was used.

A.3 File Locations and Points of Interest

All relevant files are located in the hierarchy inside the *Assets* folder. The scripts that were developed for this program can be found inside the *Assets**Scripts* folder. The main scene that is used is named *Robothespian.unity* and can be found in *Assets**Scenes*. The implementation of the sphere where the video is projected is named *inside-sphere* and can be found in *Assets* with a ready to use Unity prefab in the same location.

Appendix B

User manual

B.1 Launching the Software

The Application can either be started as a standalone build or through Unity.

B.2 Key Bindings

The functions of the application, apart from the operation of the robot, can be controlled through pressing keys on the keyboard. The full set of key bindings is listed in table B.1.

B.3 Calibrating the Avatar

When the avatar is ready to be calibrated to match the operator's measurements, the operator assumes the T-Pose and the Experimenter presses the assigned hotkey (C). It can be pressed again if the measurements seem incorrect.

B.4 Operating the Robot

When it is time for the operator to directly control the robot and start the visual feed from the camera, the experimenter presses the assigned hotkey (O). There can be no return to the calibration phase without restarting the program.

C	Runs the script that calibrates the avatar's size.
O	Enables the module that allows the user operate Robothespian.

Table B.1: Mapping of functions of the application to keys on the keyboard.

Appendix C

Pilot Study Results

There are results that add to the value of our report, but would not fit within the limits of the main report, namely the graphs indicating each participant's performance in the Pilot Study. Additionally, each participant's answers to the questionnaire are shown at Table D.2, with numeric choices that are explained in Table C.1. Finally, Table D.1 how each participant reacted to the "Instinctive Reaction" stage.

Number	Explanation
0	Strongly disagree
1	Somewhat disagree
2	Disagree
3	Neither agree nor disagree
4	Agree
5	Somewhat agree
6	Strongly agree

Table C.1: Questionnaire's numeric answers explanation.

Participant	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11	Q12	Q13	Q14	Q15
1	0	4	0	0	4	1	1	1	1	1	3	4	4	1	4
2	5	1	1	5	4	1	2	5	4	1	0	6	5	1	1
3	4	2	5	0	5	5	2	4	5	4	0	5	5	1	6
4	6	0	0	5	4	6	3	6	0	0	0	5	4	5	4

Table C.2: Pilot study's participant responses to the Questionnaire

Participant	Reaction
1	Turned Left
2	Confused
3	Turned Right
4	Turned Right

Table C.3: Pilot study's participant responses "Instinctive Reaction" stage.

C.1 Participant 1

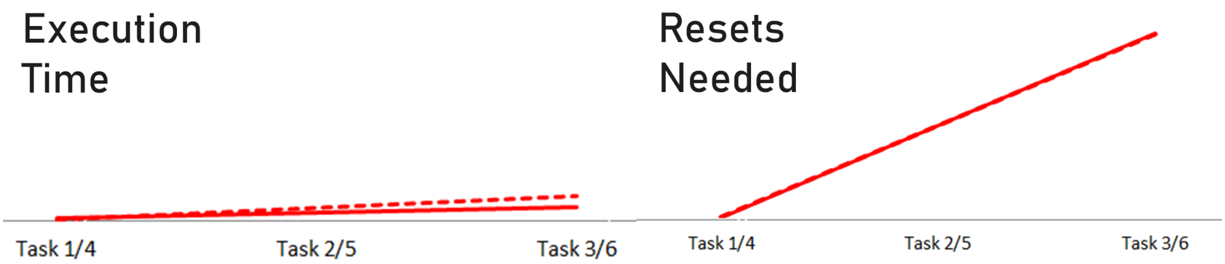

Figure C.1: Pilot study's participant 1's performance trends change in "Grabbing Objects" task.

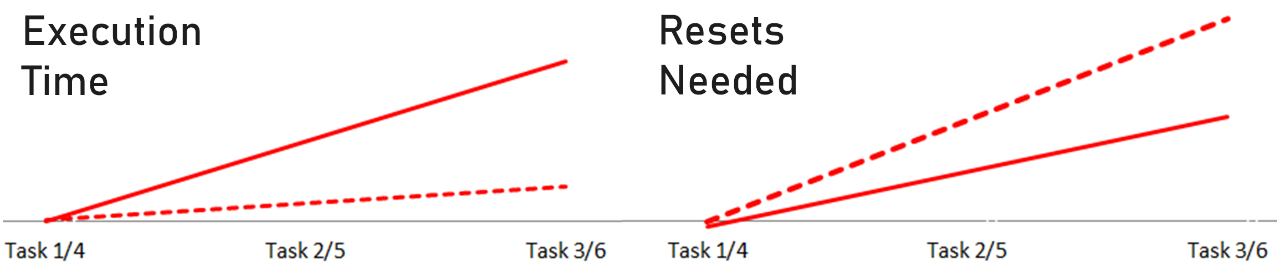

Figure C.2: Pilot study's participant 1's performance trends change in "Shape Matching" task.

C.2 Participant 2

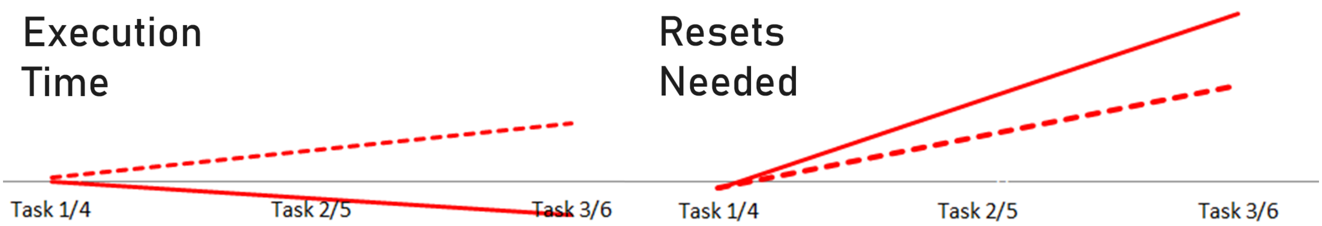

Figure C.3: Pilot study's participant 2's performance trends change in "Grabbing Objects" task.

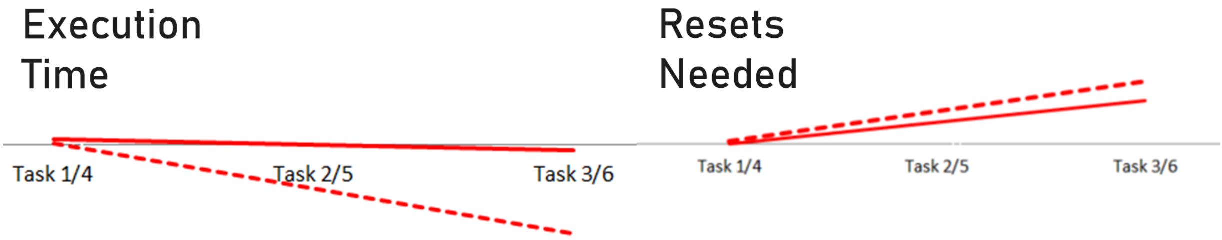

Figure C.4: Pilot study's participant 2's performance trends change in "Shape Matching" task.

C.3 Participant 3

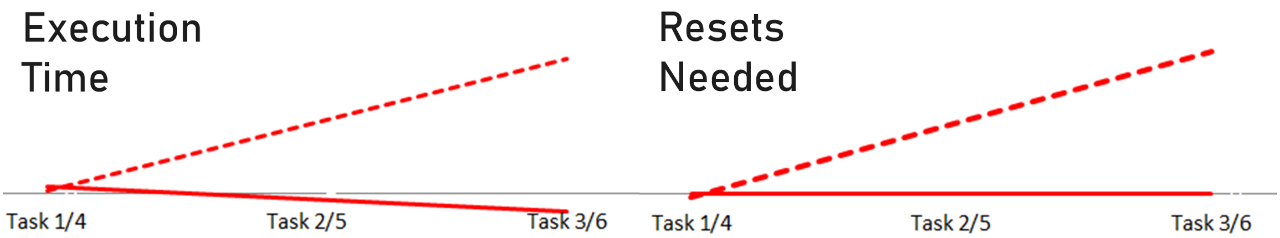

Figure C.5: Pilot study's participant 3's performance trends change in "Grabbing Objects" task.

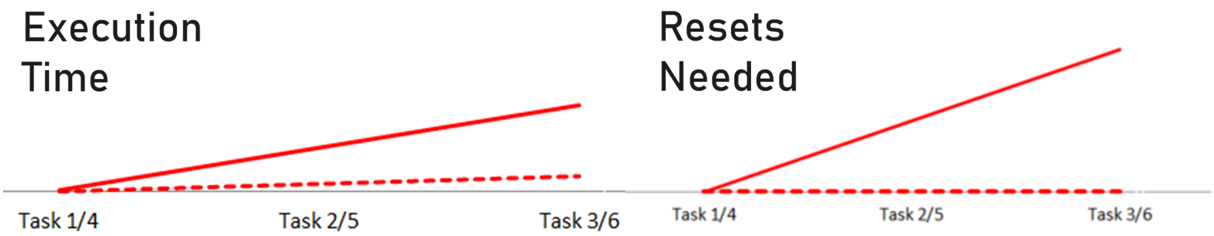

Figure C.6: Pilot study's participant 3's performance trends change in "Shape Matching" task.

C.4 Participant 4

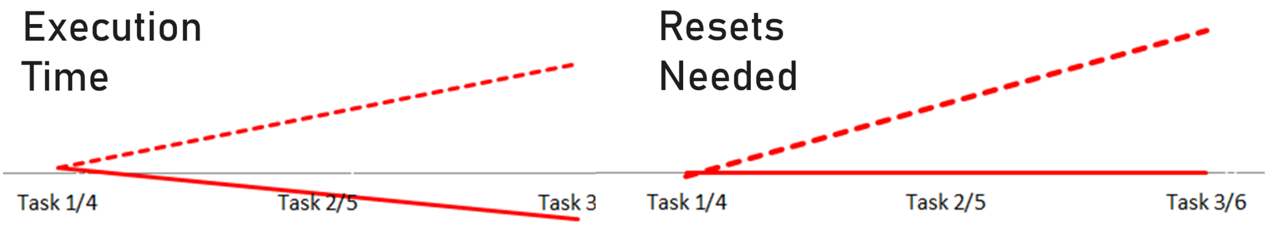

Figure C.7: Pilot study's participant 4's performance trends change in "Grabbing Objects" task.

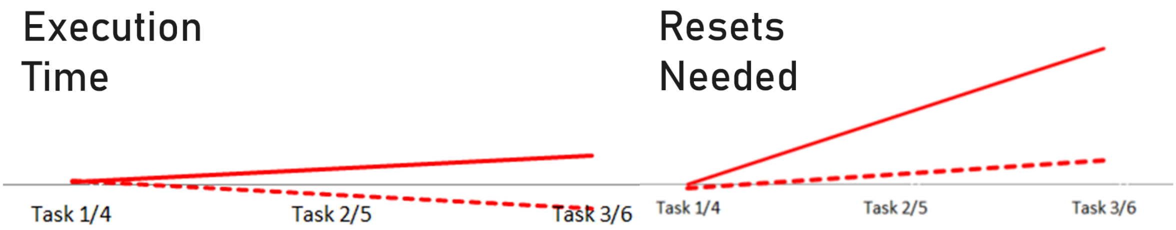

Figure C.8: Pilot study's participant 4's performance trends change in "Shape Matching" task.

Appendix D

User Study Results

Same as C, but regarding the User Study.

Participant	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Q11
1	4	4	2	1	4	2	5	5	5	1	6
2	4	4	5	2	5	4	5	6	5	4	5
3	4	2	5	0	6	4	5	5	5	5	5
4	5	4	5	1	5	4	5	4	5	6	5
5	2	3	4	4	2	2	4	4	4	4	4
6	4	5	6	0	5	5	5	3	5	5	5
7	5	2	5	1	1	4	4	4	1	5	5
8	4	4	5	4	5	3	4	6	4	6	5
9	2	5	6	0	4	5	4	5	4	4	4
10	4	2	5	3	6	4	4	2	4	6	5
11	1	2	2	5	4	2	3	1	4	6	4
12	3	6	6	0	6	5	6	6	6	6	6
13	2	4	5	1	5	4	4	3	4	4	4
14	4	1	6	3	6	5	5	5	5	5	5
15	2	5	6	1	5	5	6	6	6	6	5
16	4	5	6	1	4	5	5	3	5	3	5
17	5	5	5	2	5	5	4	5	4	4	5
18	5	5	6	1	6	5	4	4	5	5	5
19	5	5	5	1	5	5	5	0	4	6	6
20	4	4	5	2	5	5	5	3	5	4	5

Table D.1: User study's participant responses to the Questionnaire

Participant	Reaction
1	Turned Left
2	Confused
3	Turned Right
4	Turned Right
5	Turned Left
6	Turned Right
7	Turned Right
8	Turned Right
9	Turned Right
10	Turned Right
11	Confused
12	Turned Right
13	Confused
14	Turned Right
15	Confused
16	Turned Left
17	Turned Right
18	Turned Right
19	Turned Right
20	Confused

Table D.2: User study's participant responses "Instinctive Reaction" stage.

D.1 Participant 1

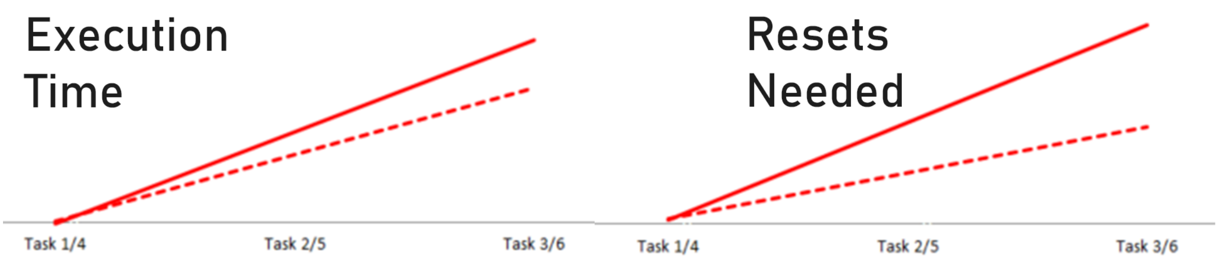

Figure D.1: User study's participant 1's performance trends change in "Grabbing Objects" task.

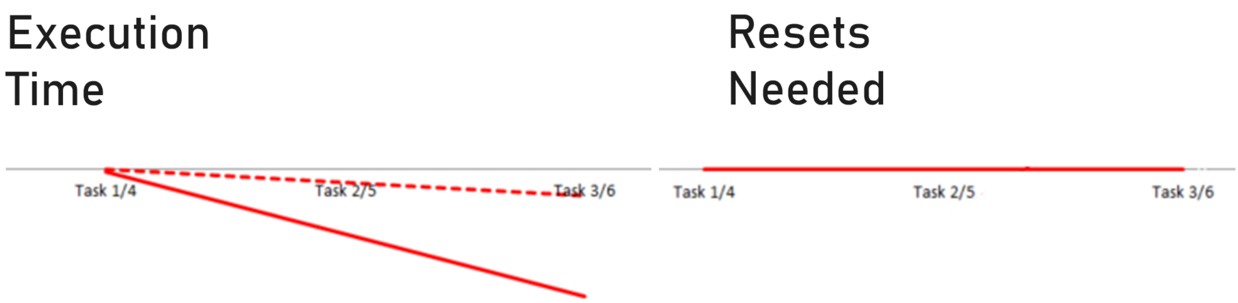

Figure D.2: User study's participant 1's performance trends change in "Shape Matching" task.

D.2 Participant 2

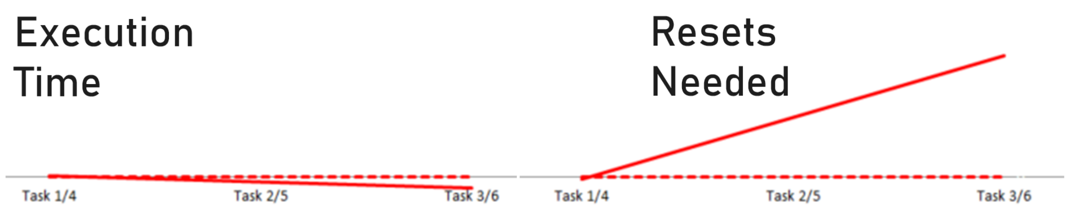

Figure D.3: User study's participant 2's performance trends change in "Grabbing Objects" task.

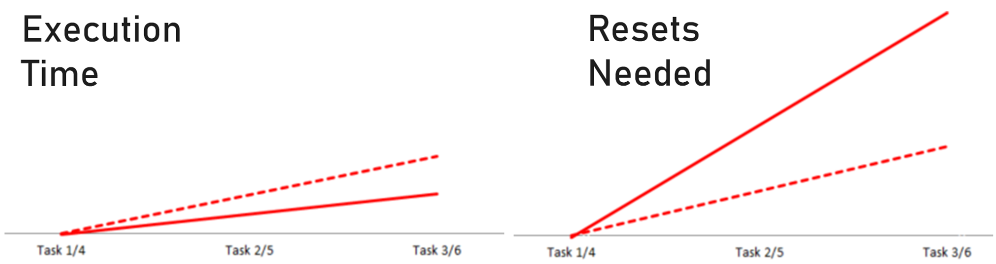

Figure D.4: User study's participant 2's performance trends change in "Shape Matching" task.

D.3 Participant 3

Figure D.5: User study's participant 3's performance trends change in "Grabbing Objects" task.

Figure D.6: User study's participant 3's performance trends change in "Shape Matching" task.

D.4 Participant 4

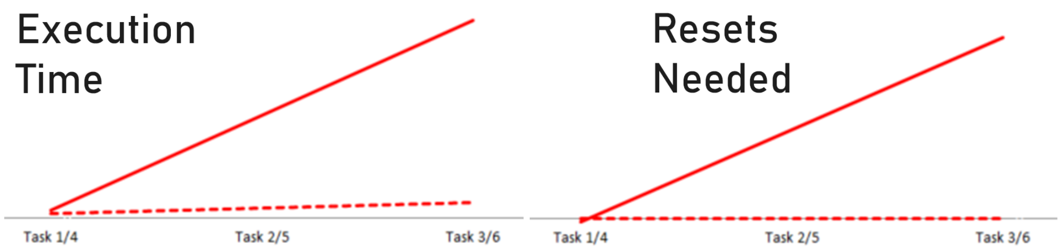

Figure D.7: User study's participant 4's performance trends change in "Grabbing Objects" task.

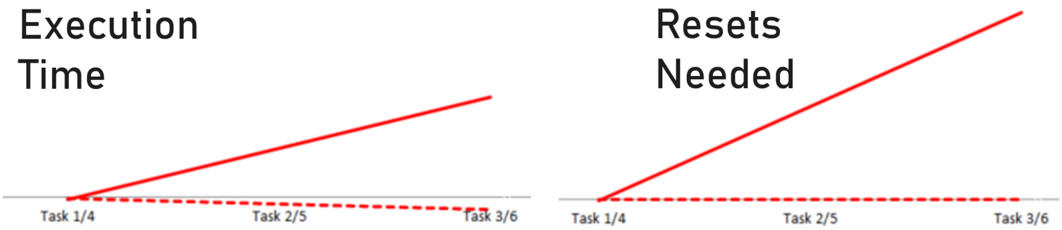

Figure D.8: User study's participant 4's performance trends change in "Shape Matching" task.

D.5 Participant 5

Figure D.9: User study's participant 5's performance trends change in "Grabbing Objects" task.

Figure D.10: User study's participant 5's performance trends change in "Shape Matching" task.

D.6 Participant 6

Figure D.11: User study's participant 6's performance trends change in "Grabbing Objects" task.

Figure D.12: User study's participant 6's performance trends change in "Shape Matching" task.

D.7 Participant 7

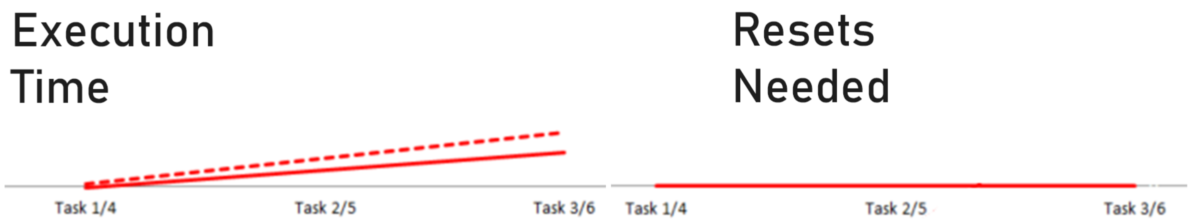

Figure D.13: User study's participant 7's performance trends change in "Grabbing Objects" task.

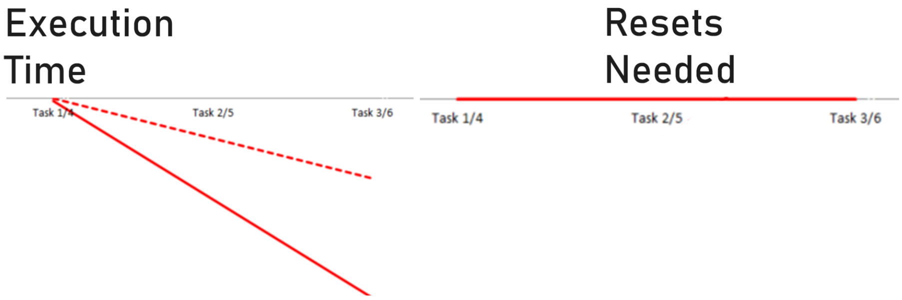

Figure D.14: User study's participant 7's performance trends change in "Shape Matching" task.

D.8 Participant 8

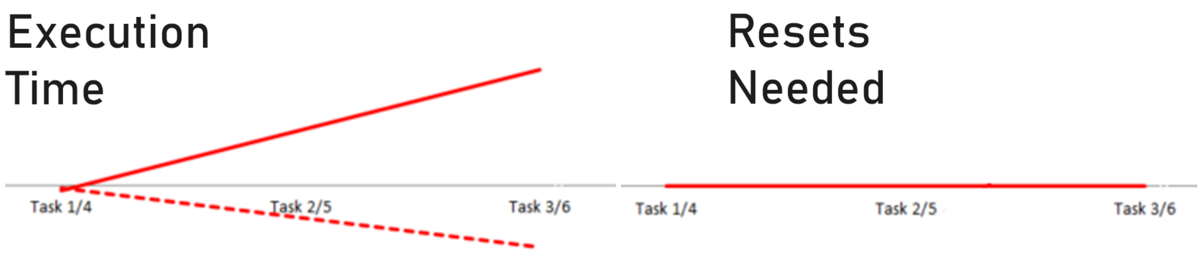

Figure D.15: User study's participant 8's performance trends change in "Grabbing Objects" task.

Figure D.16: User study's participant 8's performance trends change in "Shape Matching" task.

D.9 Participant 9

Figure D.17: User study's participant 9's performance trends change in "Grabbing Objects" task.

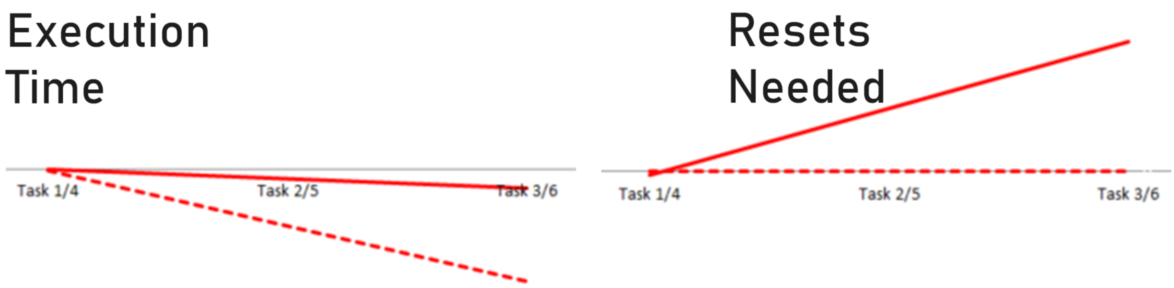

Figure D.18: User study's participant 9's performance trends change in "Shape Matching" task.

D.10 Participant 10

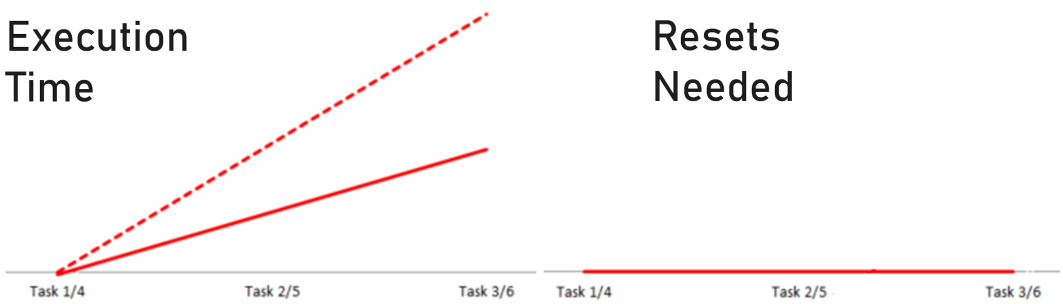

Figure D.19: User study's participant 10's performance trends change in "Grabbing Objects" task.

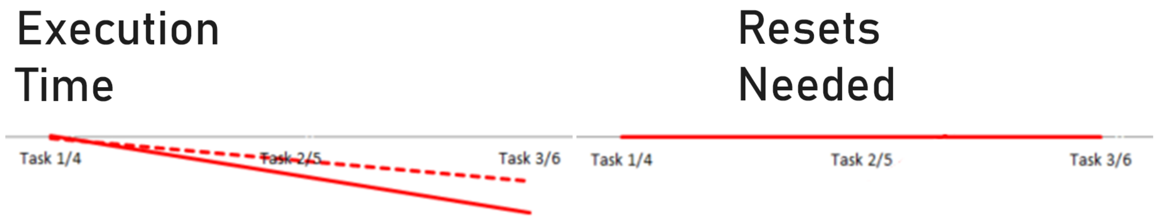

Figure D.20: User study's participant 10's performance trends change in "Shape Matching" task.

D.11 Participant 11

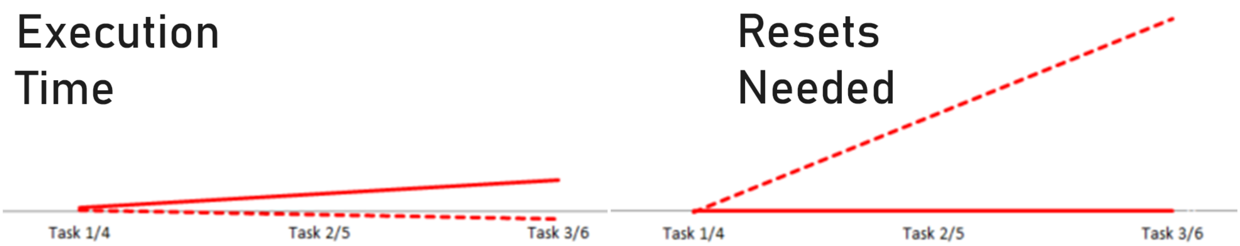

Figure D.21: User study's participant 11's performance trends change in "Grabbing Objects" task.

Figure D.22: User study's participant 11's performance trends change in "Shape Matching" task.

D.12 Participant 12

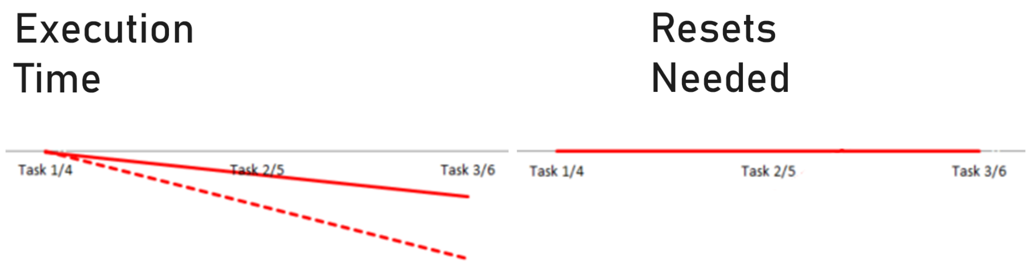

Figure D.23: User study's participant 12's performance trends change in "Grabbing Objects" task.

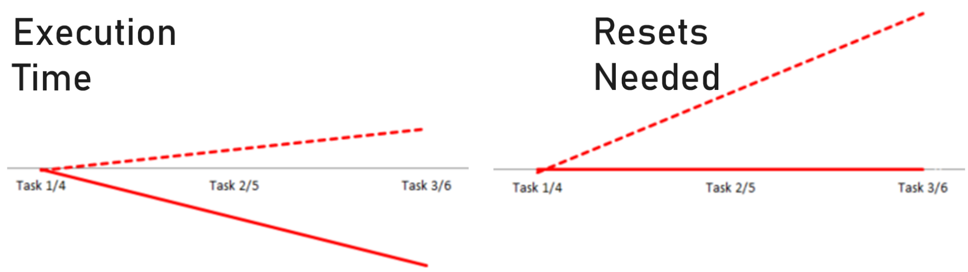

Figure D.24: User study's participant 12's performance trends change in "Shape Matching" task.

D.13 Participant 13

Figure D.25: User study's participant 13's performance trends change in "Grabbing Objects" task.

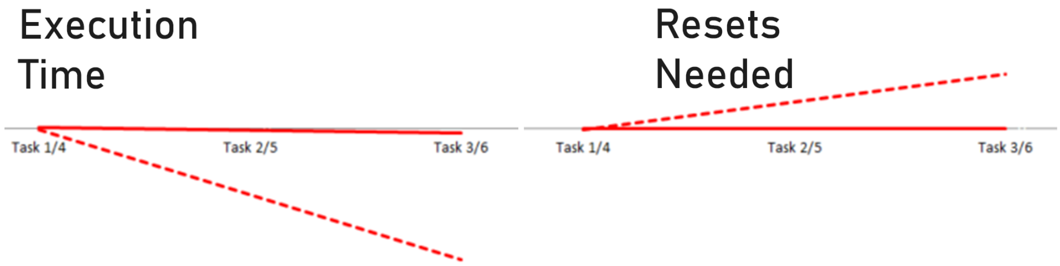

Figure D.26: User study's participant 13's performance trends change in "Shape Matching" task.

D.14 Participant 14

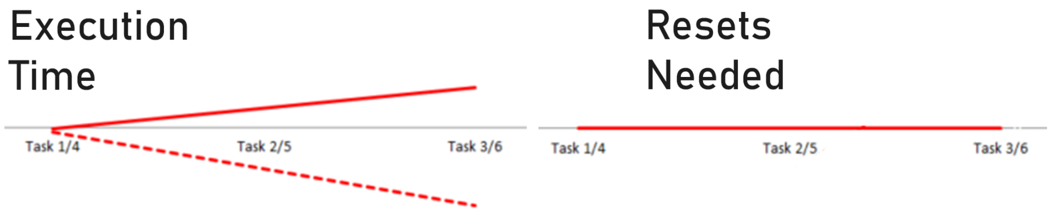

Figure D.27: User study's participant 14's performance trends change in "Grabbing Objects" task.

Figure D.28: User study's participant 14's performance trends change in "Shape Matching" task.

D.15 Participant 15

Figure D.29: User study's participant 15's performance trends change in "Grabbing Objects" task.

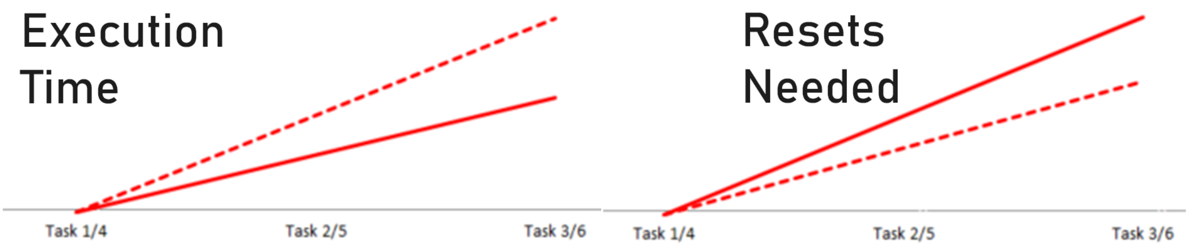

Figure D.30: User study's participant 15's performance trends change in "Shape Matching" task.

D.16 Participant 16

Figure D.31: User study's participant 16's performance trends change in "Grabbing Objects" task.

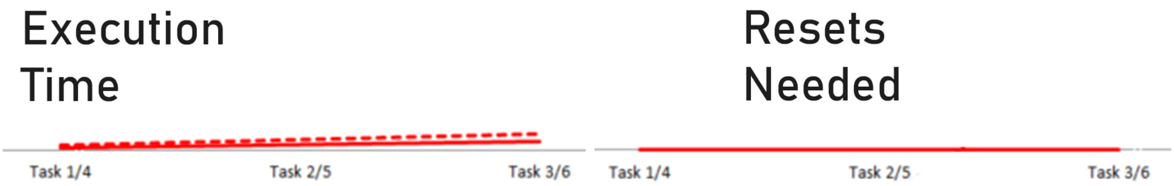

Figure D.32: User study's participant 16's performance trends change in "Shape Matching" task.

D.17 Participant 17

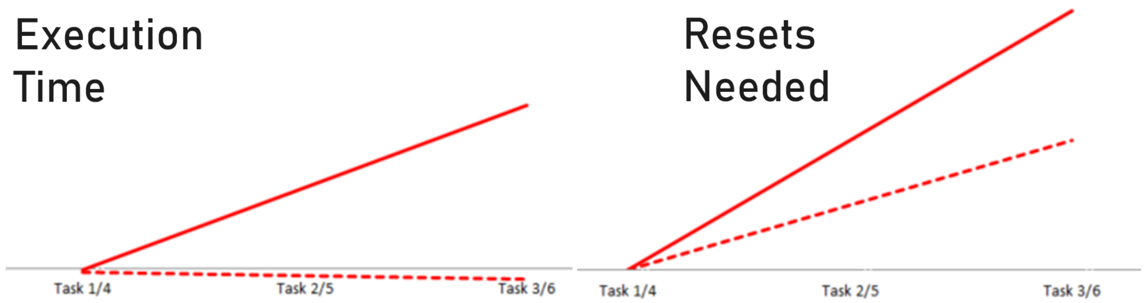

Figure D.33: User study's participant 17's performance trends change in "Grabbing Objects" task.

Figure D.34: User study's participant 17's performance trends change in "Shape Matching" task.

D.18 Participant 18

Figure D.35: User study's participant 18's performance trends change in "Grabbing Objects" task.

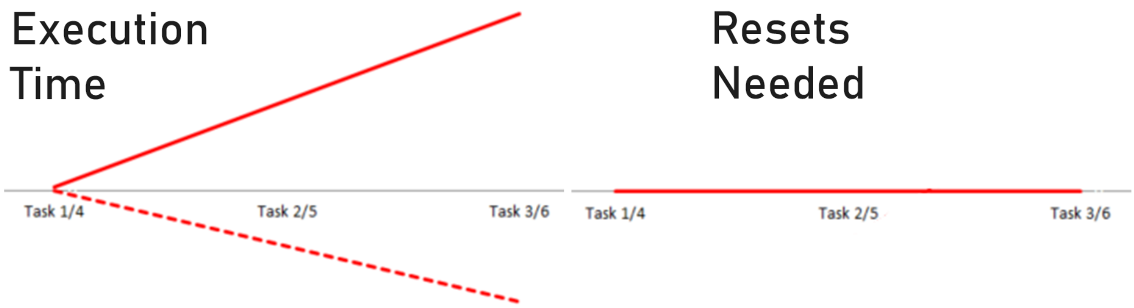

Figure D.36: User study's participant 18's performance trends change in "Shape Matching" task.

D.19 Participant 19

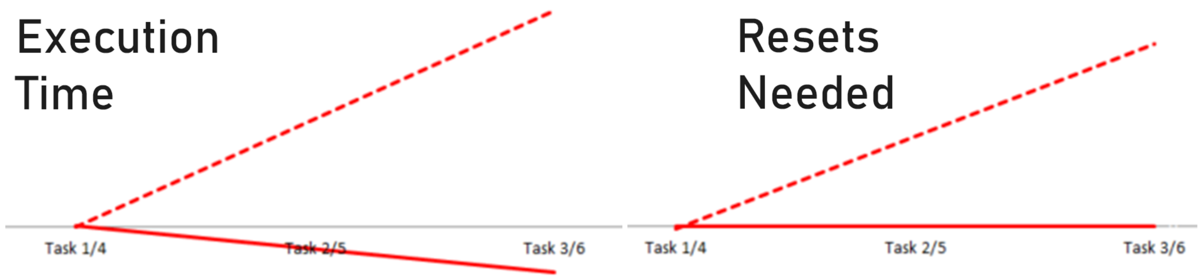

Figure D.37: User study's participant 19's performance trends change in "Grabbing Objects" task.

Figure D.38: User study's participant 19's performance trends change in "Shape Matching" task.

D.20 Participant 20

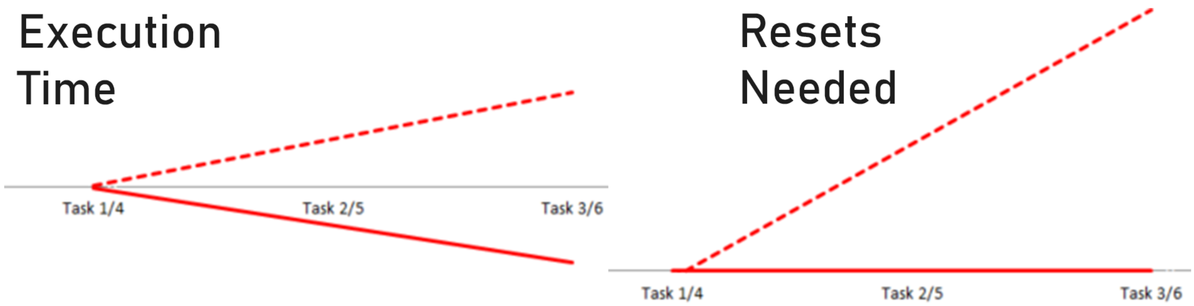

Figure D.39: User study's participant 20's performance trends change in "Grabbing Objects" task.

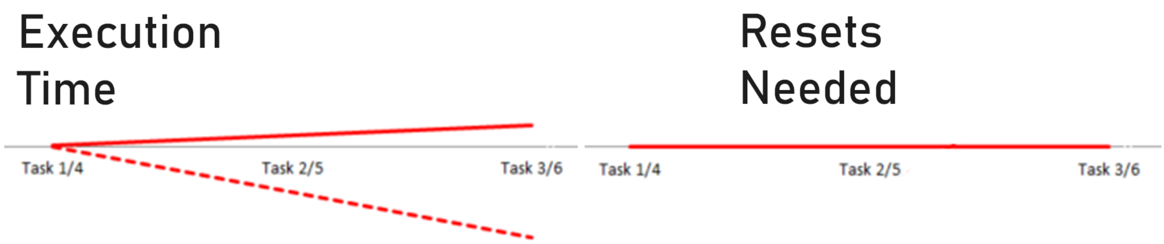

Figure D.40: User study's participant 20's performance trends change in "Shape Matching" task.

Appendix E

Code Listing

There are scripts and snippets of code that add to the value of this dissertation, but would not fit within the limits of the main report.

E.1 Tritium Code

This part is run on Robothespian's Tritium opereating framework. It handles everything on its side - receiving network messages, translating the commands, issuing them to Robothespian.

```
#Library Imports
from tritium.control import ControlFunction
from tritium.arbitration import BidType, Precedence, arbitration_by, arbitration_check
from socketserver import UDPServer, ThreadingMixIn, BaseRequestHandler
from threading import Thread
from queue import Queue

#Base Class
class UDPSequenceControlRequestHandler(BaseRequestHandler):

   command_queue = None

   def handle(self):
      data = self.request[0].strip()
      cmd = data.decode('utf-8')
      self.command_queue.put(cmd)


class ThreadedUDPServer(ThreadingMixIn, UDPServer):
   allow_reuse_address = True

# Creation of UDP Connection
class UDPSequenceControlServer(ControlFunction):

   arbitration = { #For robots with arbitration only
            'controls' : 'TTS',
            'precedence': Precedence.USER_MAXIMUM,
            'type': BidType.ON_DEMAND
   }

   HOST = ""
   PORT = 9999

   def on_activate(self):
      self.server = s = ThreadedUDPServer((self.HOST, self.PORT),
UDPSequenceControlRequestHandler)
      UDPSequenceControlRequestHandler.command_queue = self.command_queue = Queue()

      self.server_thread = t = Thread(target=s.serve_forever)
      t.daemon = True
      t.start()

   def on_deactivate(self):
      self.robot.stop_all_sequences()
      self.robot.play_sequence('rest_pose')
      self.server.shutdown()

   def on_tick(self):
      q = self.command_queue
      while not q.empty():
         cmd = q.get_nowait()
```

```
            self.debug_out('command: {}'.format(cmd))
            self.handle_command(cmd)

    def handle_command(self, cmd):

        r = self.robot
        left_ind = self.robot['Flex Left Index'].demand

        # Stop Everything
        if cmd.startswith('stop'):
            self.debug_out('STOP')
            r.stop_all_sequences()

        # Stop Everything and Reset
        elif cmd.startswith('Reset'):
            self.debug_out('RESET')
            self.relax()

        # Play Sequence
        elif cmd.startswith('play:'):
            sn = cmd[5:]
            self.debug_out('PLAY {0}'.format(sn))
            r.play_sequence(sn)

        # Fist Controls
        # Left Fist
        elif cmd.startswith('LFist:'):
            sn = cmd[6:]
            if(sn=="0"):
                #No Fist, Open Left Fist
                r['Thumb Pitch Left'].demand = 0
                r['Thumb Roll Left'].demand = 0
                r['Thumb Flex Left'].demand = 0
                r['Flex Left Index'].demand = 0
                r['Flex Left Middle'].demand = 0
                r['Flex Left Pinky'].demand = 0
                r['Flex Left Ring'].demand = 0
            else:
                #Open Left Fist
                r['Thumb Pitch Left'].demand = 4800
                r['Thumb Roll Left'].demand = 4800
                r['Thumb Flex Left'].demand = 4800
                r['Flex Left Index'].demand = 4800
                r['Flex Left Middle'].demand = 4800
                r['Flex Left Pinky'].demand = 4800
                r['Flex Left Ring'].demand = 4800
        # Right Fist
        elif cmd.startswith('RFist:'):
            sn = cmd[6:]
            if(sn=="0"):
                #No Fist, Open Right Fist
```

```
            r['Thumb Pitch Right'].demand = 0
            r['Thumb Roll Right'].demand = 0
            r['Thumb Flex Right'].demand = 0
            r['Flex Right Index'].demand = 0
            r['Flex Right Middle'].demand = 0
            #r['Flex Right Pinky'].demand = 0
            r['Flex Right Ring'].demand = 0
        else:
            #Open Right Fist
            r['Thumb Pitch Right'].demand = 4800
            r['Thumb Roll Right'].demand = 4800
            r['Thumb Flex Right'].demand = 4800
            r['Flex Right Index'].demand = 4800
            r['Flex Right Middle'].demand = 4800
            #r['Flex Right Pinky'].demand = 4800
            #r['Flex Right Ring'].demand = 4800

    # Torso Control
    # Input: Torso:-10+10
    elif cmd.startswith('Torso:'):
        sn = cmd[6:]
        self.debug_out('Torso {0} {1}'.format(sn[:3],sn[-3:]))
        #L Arm Up [-10,10]: Local Rotation Z
        r['Torso Pitch'].demand = int(sn[:3])
        #L Arm Twist [-15,15]: Local Rotation X
        r['Torso Yaw'].demand = int(sn[-3:])

    # Left Upper Arm Control
    # Input: LArm:-10-20+30
    elif cmd.startswith('LArm:'):
        sn = cmd[5:]
        self.debug_out('Left Arm Up {0} {1} {2}'.format(sn[:3],sn[3:6],sn[-3:]))
        #L Arm Up [-80,0]: Local Rotation Z
        r['Shoulder Pitch Left'].demand = int(sn[:3])
        #L Arm Out [-80,-20]: Local Rotation Y
        r['Shoulder Roll Left'].demand = int(sn[3:6])
        #L Arm Twist [-25,35]: Local Rotation X
        r['Shoulder Yaw Left'].demand = int(sn[-3:])

    # Left Forearm Control
    elif cmd.startswith('LForearm:'):
        sn = cmd[9:]
        self.debug_out('Left Forearm Up {0} {1}'.format(sn[:3],sn[-3:]))
        #L Arm Up [-80,10]: Local Rotation Z
        r['Elbow Pitch Left'].demand = int(sn[:3])
        #L Wrist Roll [-80,80]: Local Rotation Y
        r['Wrist Roll Left'].demand = int(sn[-3:])

    # Right Upper Arm Control
    # Input: RArm:-10-20+30
    elif cmd.startswith('RArm:'):
```

```
            sn = cmd[5:]
            self.debug_out('Right Arm Up {0} {1} {2}'.format(sn[:3],sn[3:6],sn[-3:]))
            #L Arm Up [-80,-10]: Local Rotation Z
            r['Shoulder Pitch Right'].demand = int(sn[:3])
            #L Arm Out [-85,-35]: Local Rotation Y
            r['Shoulder Roll Right'].demand = int(sn[3:6])
            #L Arm Twist [-25,35]: Local Rotation X
            r['Shoulder Yaw Right'].demand = int(sn[-3:])

        # Right Forearm Control
        elif cmd.startswith('RForearm:'):
            sn = cmd[9:]
            self.debug_out('Right Forearm Up {0} {1}'.format(sn[:3],sn[-3:]))
            #R Arm Up [-80,10]: Local Rotation Z
            r['Elbow Pitch Right'].demand = int(sn[:3])
            #R Wrist Roll [-80,10]: Local Rotation Y
            r['Wrist Roll Right'].demand = int(sn[-3:])

        else:
            self.debug_out('WARNING unrecognised command: {0}'.format(cmd))

    def relax(self):
            self.robot.stop_all_sequences()
            self.robot['Shoulder Pitch Left'].demand = -80
            self.robot['Shoulder Roll Left'].demand = -85
            self.robot['Shoulder Yaw Left'].demand = 5
            self.robot['Elbow Pitch Left'].demand = -80
            self.robot['Wrist Pitch Left'].demand = 0
            self.robot['Wrist Roll Left'].demand = 10

            self.robot['Shoulder Pitch Right'].demand = -80
            self.robot['Shoulder Roll Right'].demand = -85
            self.robot['Shoulder Yaw Right'].demand = 5
            self.robot['Elbow Pitch Right'].demand = -80
            self.robot['Wrist Pitch Right'].demand = 0
            self.robot['Wrist Roll Right'].demand = 10


            self.robot['Torso Pitch'].demand = 0
            self.robot['Torso Roll'].demand = 0
            self.robot['Torso Yaw'].demand = 0

            self.robot['Head Pitch'].demand = 0
            self.robot['Head Roll'].demand = 0
            self.robot['Head Yaw'].demand = 0

            self.robot['Flex Right Index'].demand = 0
            self.robot['Flex Right Middle'].demand = 0
            self.robot['Flex Right Ring'].demand = 0
            self.robot['Flex Right Pinky'].demand = 0
            self.robot['Thumb Flex Right'].demand = 2500
```

```
self.robot['Thumb Pitch Right'].demand = 2500
self.robot['Thumb Roll Right'].demand = 4800
self.robot['Finger Spread Right'].demand = 2500

self.robot['Flex Left Index'].demand = 0
self.robot['Flex Left Middle'].demand = 0
self.robot['Flex Left Ring'].demand = 0
self.robot['Flex Left Pinky'].demand = 0
self.robot['Thumb Flex Left'].demand = 2500
self.robot['Thumb Pitch Left'].demand = 2500
self.robot['Thumb Roll Left'].demand = 4800
self.robot['Finger Spread Left'].demand = 2500

self.robot.play_sequence('RestPose')
```

E.2 Unity UDP Script

This script is responsible for handling and sending messages to Robothespian's Tritium.

```
using UnityEngine;
using UnityEngine.Networking;
using UnityEngine.Networking.NetworkSystem;
using System;
using System.Text;
using System.IO;
using System.Net;
using System.Net.Sockets;
using System.Threading;

public class UDPSend : MonoBehaviour
{
    public static string serverIP;
    public static string text;
    public static string txtfile;
    public static int count = 0;
    public static int serverPort;
    public static bool debug;

    void Start()
    {
        serverIP = "192.168.32.2";
        serverPort = 9999;
        debug = true;
    }

    //Send function
    public void ClientSend(string category, int angles)
    {
        text = category + angles;
        byte[] packetdata = Encoding.ASCII.GetBytes(text);
        IPEndPoint ep = new IPEndPoint(IPAddress.Parse(serverIP), serverPort);
        Socket sock = new Socket(AddressFamily.InterNetwork, SocketType.Dgram,
ProtocolType.Udp);

        try
        {
            sock.SendTo(packetdata, ep);
            //Thread.Sleep(50);
        }
        catch(Exception ex)
        {
            if (debug) Debug.Log("<color=red>EXCEPTION COUGHT ON SENDING:</color>" +
text);
        }
    }

    //Send function
    public void ClientSend(string category, string message)
    {
        text = category + message;
        byte[] packetdata = Encoding.ASCII.GetBytes(text);
        IPEndPoint ep = new IPEndPoint(IPAddress.Parse(serverIP), serverPort);
        Socket sock = new Socket(AddressFamily.InterNetwork, SocketType.Dgram,
ProtocolType.Udp);

        try
        {
            sock.SendTo(packetdata, ep);
            //Thread.Sleep(50);
        }
        catch (Exception ex)
```

```
        {
            if (debug) Debug.Log("<color=red>EXCEPTION COUGHT ON SENDING:</color>" +
text);
        }
    }
}
```

E.3 Unity Rotation Stabilizer Script

This script is responsible for stabilizing the trackers used in the avatar's limbs. It counteracts the rotation that is caused by the operator.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class RotationStabilizerTwo : MonoBehaviour
{
    private GameObject parent;
    private Quaternion rotation, prev_parent_rot, deltaRotation;

    void Awake()
    {
        rotation = transform.rotation;
    }

    // Start is called before the first frame update
    void Start()
    {
        parent = this.transform.parent.gameObject;
        prev_parent_rot = parent.transform.rotation;
    }

    // Update is called once per frame
    void Update()
    {
            deltaRotation = parent.transform.rotation * 
Quaternion.Inverse(prev_parent_rot);
            this.transform.rotation = Quaternion.Inverse(deltaRotation) * 
this.transform.rotation;
            prev_parent_rot = parent.transform.rotation;
    }
}
```

Appendix F

Project Plan

F.1 Outline

In recent years, there has been a noticeably rapid developments in **telerobotics** technologies. Telerobotics, likewise called teleoperation, is the specialized technical term for the remote manipulation of a complex robot, named **"telechir"**. In a telerobotic framework, a human operator dictates the actions of a telechir, which could be as simple as a single robotic arm or as refined as an humanoid robot (android). This control is achieved through signals exchanged between the operator's interface and the telechir's central processor unit (CPU) and are called **"telemetry"**.

These rapid developments go hand in hand with the revitalized emergence of Virtual and Augmented Reality technologies. While telerobotics was usually achieved by using primitive control interfaces, like joysticks, it can now be easily achieved through the use of a virtual reality off the shelf system. Thus, telerobotics is entering a new spectrum, that of Virtual and Augmented controlled telechirs, and has introduced a more refined type of teleoperation known as **"telepresence"**. By utilizing a telepresence framework the human operator achieves an experience akin to Virtual Reality, meaning that through sensory input, like vision and sound, from the telechir, he has the sense of actually being on the remote location.

The presented telepresence setup will utilize the required built-in and supplementary sensors of the **Robothespian** robot, in order to allow the operator to feel as immersed as possible, while maintaining a high degree of functionality.

F.2 Aims and Objectives

The aim of the project is to investigate the effectiveness of operating a robot remotely using our designed software and sensor setup. This will be evaluated through a user study with a set of tasks to be performed while operating the robot.

F.2.1 Expected project scope

The telepresence system should offer the operator basic control of the robot. He will be able to move his view around, with the lowest latency we can achieve, which is important in Virtual Reality. He will also be able to move his, and therefore the robot's, arms freely.

Due to a wide range of reasons, from hardware speed restrictions to input latency, the operator might experience certain inconsistencies between his movements and those of the telechir's. That would especially be the case in situations where he attempts to move faster than the hardware permits. To deal with these cases, he will be able to see in his Virtual Interface an approximation of where his real hands are compared to the telechir's. Then, based on their distance, the robot would either trail behind them or stop interacting entirely until the operator synchronizes their positions again.

F.2.2 Extended project scope

Telechir's perception as a person could be augmented by expanding the head viewing mechanics. When the user moves his gaze to something that is not in the direct gaze of the robot, the head could move, trying to mirror the position and orientation of the operator's. That is forbidding in some cases due to hardware restrictions, like when looking down, but in cases where the visual target is sideways it could be possible.

F.2.3 Evaluation

In order to evaluate the usability - functionality of the developed program, a User Study will be conducted. A small group of participants will fill the Simulation Sickness Questionnaire before and after operating Robothespian, in order to determine the quality of the virtual experience that we created. During operation of the telechir they will be asked to perform a series of tasks, from touching objects to following patterns with their hands and evaluating how well the robot tracked and executed their commands.

F.3 Time Plan

Starting from December 10th 2018, this project is expected to finish on September 6th 2019 with the submission of the Final Report and a demonstration of the complete project. In order to achieve the expected research and development objectives, a preliminary milestone plan has been drafted.

The main milestones of this project are listed below, along with their projected end dates:

- Project Plan Submission - 18/01/2019
- Hardware Setup - 18/02/2019

  During Hardware setup, the basic assembly of the additional hardware will be performed, along with installation of their required software, like their drivers.

- Software Development - 08/03/2019

  During Software Development, a Unity program will be written that controls a virtual robot that follows the same restrictions as its real world counterpart.

- Hardware and Software Integration - 22/03/2019

  During the Integration phase, a Transmission Control Protocol (TCP) server will be created that enables the communication between the real and virtual Robothespians' individual operating frameworks. We will also combine the Hardware inputs with our basic software program.

- Project Setup Completion - 01/04/2019
- User Study - Pending Approval - June/July
- (Second Optional) User Study - Pending Approval - June/July

- Final Report and Demonstration - 06/09/2019

The real end dates are flexible and may change depending on the complexity of the tasks and the issues that might arise. In order to achieve the expected objectives, the estimated workload is 222 man-hours during research period from December 10th 2018 to April 1st 2019, expecting to spend two hours on average per day on the project.

F.4 Risks

Unfortunately, there are some prospects that might range from proving detrimental to completely putting to the project to a halt.

- In the case of Robothespian suffering major mechanical failure, it would have to be sent back to Engineered Arts for repairs for an unknown amount of time.

- In case of less serious mechanical failures, for example in the servo of a joint, its functionality might be reduced to a point that is incompatible to this research's scope. As an example, a failure in the finger joints would make them unresponsive, thus making grabbing objects with the telechir unreliable at best.

F.5 Hardware Tools

As per the current framework's specifications, our project has two discrete hardware components, the Robothespian itself and a 360° camera.

F.5.1 Robothespian

Robothespian, as the name suggests, is a complex humanoid ”acting” robot designed for human interaction in a public environment. Being the prime, and most iconic, humanoid designed by Engineered Arts, it can perform an unbelievably expressive scope of movements, making it ideal for our project. It runs on a proprietary Python framework, which will - ideally - be able to interface with our software via a TCP connection.

F.5.2 360° Camera

Robothespian is not equipped with a camera capable of delivering true Virtual Reality experience. In order to provide the operator with a real-time, immersive experience, we need to attach a 360° Camera to the head of our robot, in order to preserve his feeling of scaling. The selection of the camera takes into account factors such as its size, ease of attachment, live stream resolution, Field of View and, of course, the project's budget. The mounting to the head will happen either with an off the shelf or 3D printed strap.

F.6 Software Tools

As per the current framework's specifications, the only additional software that will be utilized is Unity Real-Time Engine.

F.6.1 Unity

Unity provides a ready-to-use game engine and flexible real-time tools. Its popularity and extensive documentation allow for many off-the-shelve solutions for both assets and issues that may arise. By using its advanced Rigging system, we are able to create a basic bone rig that follows the join angle limitations that Robothespian's manufacturer has set, effectively making a Virtual Reality controlled Robothespian version in Unity, from now called **"Unity's Robothespian"**. For synchronizing Robothespian's movement with Unity's, we will utilize Unity Engine's API, which allows us to create a TCP Client-Server Connection, achieving bidirectional communication between their frameworks.

F.6.2 Tritium

Tritium is the Engineered Arts' in-house developed framework that their robots use. It offers both a web and direct command input based interfaces for an operator. It is designed so a robots functionality can be built up in little software blocks, or nodes, which can contain code from almost any programming language. As mentioned before, a basic aspect of this project is creating a bidirectional communication between Tritium and Unity, which will be achieved by sending C# commands from Unity through our server to Tritium, which will translate them in understandable commands.

Appendix G

Original Protocol

Experiment Protocol – Robothespian

Experimenter Instruction

Actions

0: Introduction – Forms - Demographic
“Thank you for coming. I am Nick. First, let’s start with the paperwork. This is the Information Sheet. It contains information regarding this experiment. It also includes that you can abort the experiment at any point without giving us a reason. Please read it carefully. This is the consent form. In order to participate you have to read it, tick the boxes and sign it. Please do it twice; one copy is for you, one is for us. If there are any questions, now or through the course of the experiment, please ask us. The last paper is a short survey that serves to give us an impression of your previous experiences with virtual reality.”

1: Demonstration of Program - Robothespian
“This is the virtual reality headset and controllers you will be using. You will face this wall while controlling a virtual avatar that is connected to this robot. Before we begin, I would like to show you the limits of the Robot’s limbs. While it appears humanoid, its limbs work in a particular way. I will run the demo program to display the ranges.”

<Experimenter executes the program that shows each movement’s limits, while explaining.>

“Now, I would like to show you your Heads-Up display. These depictions on the upper right corner show your avatar’s movement. If you move outside the range of the allowed ranged in any way, you will see that there will be a red avatar moving as you move, while a green avatar will be still. If you want the robot to follow your order, you should move the red parts towards the green. Do you have any questions?”

2: Virtual Avatar acclimation stage
“Good. Now, let me explain the controls. The trackpad and the grip buttons control the fist controls of each hand. Press either once to close the robot’s fist, and again when you want to open it. These controls will work when you actively control the robot. Let’s get you into the virtual world.”

<Help participant into the VR headset>

“Good. Now please stand straight and extend your arms to the side.”

<Experimenter presses “C” to calibrate>

“Good. Now try moving your arms around and see if the avatar responds to your movements. Keep in mind that you will observe no Torso and Head rotation or movement on the avatar, since the robot does not support them, so try to stay in the approximate spot if you can. However, when you move your head around, your viewpoint changes as it normally should. You have 5 minutes to adjust to the movement mapping. Tell me if you are ready before the 5 minutes pass and want to proceed to the next step.”

<Participant fiddles around with his arms.>

“Good. Now let’s give you control of the robot. While the robot cannot move its head, you are able to see around you using the attached camera. Like before, you have 5 minutes to adjust to the movement mapping, hand controlling and the depth perception. You may play with anything you want on the table, I will reset them. Like before, tell me if you are optimistic about your grasp of the controls and want us to move to the first test."

<Participant fiddles with the robot.>

3: Grasping Stage

"Let us move to the first task. You will use your left hand to grasp this cylinder <Show red cylinder> and you will drop it in this box <Shows open box>. You will perform this task 6 times. Any questions?"

<Experimenter places the cylinder on the table and the box on its spot.>

"Go when ready"

<Experimenter times and notes participant's performance.>

4: Matching Shapes Stage

"Let us move to the next task. I will now place the cylinder in your hand in horizontal position and you will grasp it. Then, you have to drop it in this box, through this hole <Shows hole in the box's cover>. You will perform this task 6 times. Any questions?"

<Experimenter acts accordingly.>

"Go when ready"

<Experimenter times and notes participant's performance.>

5: Embodiment Stage

"Let us move to the final task. I want to test the movement capabilities of the robot combined with the camera. I will give you some verbal commands, like look at your right hand, and you will have to execute them as fast as possible. Any Questions?"

<Experimenter commands: Look at your Right Hand, Look at the first box, look at the wall, look at the monitor, look at your left hand, look at the cylinder, look at the bottle, look at the ceiling, **look at me.**>

6: Epilogue

"That was it for today, let me help you out of the headset."

<The experimenter helps the participant remove the headset.>

"Do you have any further comments about your experience?"

<The experimenters note down any remarks the participant may has.>

"Thank you very much for participating. There is only one last thing to do: please fill out this form so we can give you your compensation and this short Questionnaire regarding your experience. Please make sure that the address is valid, otherwise we can't get it back from the department."

<The Participant fills the payment form and the experimenters hand out the compensation.>

"Thank you again for your participation"

<The Participant leaves>

Appendix H

Revised Protocol

Experiment Protocol – Robothespian

Experimenter Instruction

Actions

0: Introduction – Forms - Demographic

"Thank you for coming. I am Nick. First, let's start with the paperwork. This is the Information Sheet. It contains information regarding this experiment. It also includes that you can abort the experiment at any point without giving us a reason. Please read it carefully. This is the consent form. In order to participate you have to read it, tick the boxes and sign it. Please do it twice; one copy is for you, one is for us. If there are any questions, now or through the course of the experiment, please ask us. The last paper is a short survey that serves to give us an impression of your previous experiences with virtual reality."

1: Demonstration of Program - Robothespian

"This is the virtual reality headset and controllers you will be using. You will face this wall while controlling a virtual avatar that is connected to this robot. Before we begin, I would like to show you the limits of the Robot's limbs. While it appears humanoid, its limbs work in a particular way. I will run the demo program to display the ranges."

<Experimenter executes the program that shows each movement's limits, while explaining.>

"Now, I would like to show you your Heads-Up display. These depictions on the upper right corner show your avatar's movement. If you move outside the range of the allowed ranged in any way, you will see that there will be a red avatar moving as you move, while a green avatar will be still. If you want the robot to follow your order, you should move the red parts towards the green. Do you have any questions?"

2: Virtual Avatar acclimation stage

"Good. Now, let me explain the controls. The trackpad and the grip buttons control the fist controls of each hand. Press either once to close the robot's fist, and again when you want to open it. These controls will work when you actively control the robot. Let's get you into the virtual world."

<Help participant into the VR headset>

"Good. Now please stand straight and extend your arms to the side."

<Experimenter presses "C" to calibrate>

"Good. Now try moving your arms around and see if the avatar responds to your movements. Keep in mind that you will observe no Torso and Head rotation or movement on the avatar, since the robot does not support them, so try to stay in the approximate spot if you can. However, when you move your head around, your viewpoint changes as it normally should. You have 5 minutes to adjust to the movement mapping. Tell me if you are ready before the 5 minutes pass and want to proceed to the next step."

<Participant fiddles around with his arms.>

"Good. Now let's give you control of the robot. While the robot cannot move its head, you are able to see around you using the attached camera. Like before, you have 5 minutes to adjust to the movement mapping, hand controlling and the depth perception. You may play with anything you want on the table, I will reset them. Like before, tell me if you are optimistic about your grasp of the controls and want us to move to the first test."

<Participant fiddles with the robot.>

3: Grasping Stage
"Let us move to the first task. You will use your left hand to grasp this cylinder <Show red cylinder> and you will drop it in this box <Shows open box>. You will perform this task 6 times. Any questions?"

<Experimenter places the cylinder on the table and the box on its spot.>

"Go when ready"

<Experimenter times and notes participant's performance.>

5: Matching Shapes Stage
"Let us move to the next task. I will now place the cylinder in your hand in horizontal position and you will grasp it. Then, you have to drop it in this box, through this hole <Shows hole in the box's cover>. You will perform this task 6 times. Any questions?"

<Experimenter acts accordingly.>

"Go when ready"

<Experimenter times and notes participant's performance.>

6: Remote Conversation Stage
"OK, for the next part we are going to have a conversation through the robot. I am going to stand in front of it and talk to you about everyday topics. Any questions?"

<Experimenter and Participant partake in small talk.>

<Experimenter moves around the robot to see how the participant responds through the conversation.>

6: Embodiment Stage
"Let us move to the final task. I want to test the movement capabilities of the robot combined with the camera. I will give you some verbal commands, like look at your right hand, and you will have to execute them as fast as possible. Any Questions?"

<Experimenter commands: Look at your Right Hand, Look at the first box, look at the wall, look at the monitor, look at your left hand, look at the cylinder, look at the bottle, look at the ceiling, **look at me.**>

7: Epilogue

"That was it for today, let me help you out of the headset."

<The experimenter helps the participant remove the headset.>

"Do you have any further comments about your experience?"

<The experimenters note down any remarks the participant may has.>

"Thank you very much for participating. There is only one last thing to do: please fill out this form so we can give you your compensation and this short Questionnaire regarding your experience. Please make sure that the address is valid, otherwise we can't get it back from the department."

<The Participant fills the payment form and the experimenters hand out the compensation.>

“Thank you again for your participation”

<The Participant leaves>

Appendix I

Original Questionnaire

Embodiment Questionnaire

Participant ID ______________ Handedness: Left-Hander / Right-Hander / Ambidextrous

Please select your level of agreement with the following statements:

"During the experiment there were moments in which…

Q1. "I felt as if the virtual body I saw when I looked down was my body"

strongly disagree	disagree	somewhat disagree	neither agree nor disagree	somewhat agree	agree	strongly agree
(-3)	(-2)	(-1)	(0)	(1)	(2)	(3)

Q2. "It felt as if the virtual body I saw was someone else"

strongly disagree	disagree	somewhat disagree	neither agree nor disagree	somewhat agree	agree	strongly agree
(-3)	(-2)	(-1)	(0)	(1)	(2)	(3)

Q3. "It seemed as if I might have more than one body"

strongly disagree	disagree	somewhat disagree	neither agree nor disagree	somewhat agree	agree	strongly agree
(-3)	(-2)	(-1)	(0)	(1)	(2)	(3)

Q4. "It felt like I could control the virtual body as if it was my own body"

strongly disagree	disagree	somewhat disagree	neither agree nor disagree	somewhat agree	agree	strongly agree
(-3)	(-2)	(-1)	(0)	(1)	(2)	(3)

Q5. "The movements of the virtual body were caused by my movements"

strongly disagree	disagree	somewhat disagree	neither agree nor disagree	somewhat agree	agree	strongly agree
(-3)	(-2)	(-1)	(0)	(1)	(2)	(3)

Q6. "I felt as if the movements of the virtual body were influencing my own movements"

strongly disagree	disagree	somewhat disagree	neither agree nor disagree	somewhat agree	agree	strongly agree
(-3)	(-2)	(-1)	(0)	(1)	(2)	(3)

Q7. "I felt as if the virtual body was moving by itself"

strongly disagree	disagree	somewhat disagree	neither agree nor disagree	somewhat agree	agree	strongly agree
(-3)	(-2)	(-1)	(0)	(1)	(2)	(3)

Q8. "I felt as if my body was located where I saw the virtual body"

strongly disagree	disagree	somewhat disagree	neither agree nor disagree	somewhat agree	agree	strongly agree
(-3)	(-2)	(-1)	(0)	(1)	(2)	(3)

Q9. “I felt out of my body”

strongly disagree	*disagree*	*somewhat disagree*	*neither agree nor disagree*	*somewhat agree*	agree	strongly agree
(-3)	*(-2)*	*(-1)*	*(0)*	(1)	(2)	(3)

Q10. “During calibration I felt as if my (real) body were drifting towards the virtual body or as if the virtual body were drifting towards my (real) body”

strongly disagree	*disagree*	*somewhat disagree*	*neither agree nor disagree*	*somewhat agree*	agree	strongly agree
(-3)	*(-2)*	*(-1)*	*(0)*	(1)	(2)	(3)

Q11. “I felt that my own body could be hit by the foam sword”

strongly disagree	*disagree*	*somewhat disagree*	*neither agree nor disagree*	*somewhat agree*	agree	strongly agree
(-3)	*(-2)*	*(-1)*	*(0)*	(1)	(2)	(3)

Q12. “I felt a sensation of fear and/or surprise in my body when I saw the foam sword”

strongly disagree	*disagree*	*somewhat disagree*	*neither agree nor disagree*	*somewhat agree*	agree	strongly agree
(-3)	*(-2)*	*(-1)*	*(0)*	(1)	(2)	(3)

Q13. “When the attack happened, I felt the instinct to protect the robot as if it was me”

strongly disagree	*disagree*	*somewhat disagree*	*neither agree nor disagree*	*somewhat agree*	agree	strongly agree
(-3)	*(-2)*	*(-1)*	*(0)*	(1)	(2)	(3)

Q14. “I had the feeling that I might be harmed by the the foam sword“

strongly disagree	*disagree*	*somewhat disagree*	*neither agree nor disagree*	*somewhat agree*	agree	strongly agree
(-3)	*(-2)*	*(-1)*	*(0)*	(1)	(2)	(3)

Q15. “If Right-Handed, did you feel like you had difficulty in controlling the Left arm?”

strongly disagree	*disagree*	*somewhat disagree*	*neither agree nor disagree*	*somewhat agree*	agree	strongly agree
(-3)	*(-2)*	*(-1)*	*(0)*	(1)	(2)	(3)

Appendix J

Revised Questionnaire

Embodiment Questionnaire

Participant ID ______________ Handedness: Left-Hander / Right-Hander / Ambidextrous

Please answer the following demographic questions:

D1. What is your age?

18 - 24 25 - 34 35 - 44 45 – 54 55 - 65

D2. How often have you used VR devices in the last year?

Never Sparsely Monthly Weekly Daily

Please select your level of agreement with the following statements:

"During the experiment there were moments in which…

Q1. "I felt as if the virtual body I saw when I looked down was my body"

strongly disagree	*disagree*	*somewhat disagree*	*neither agree nor disagree*	*somewhat agree*	agree	strongly agree
(-3)	*(-2)*	*(-1)*	*(0)*	(1)	(2)	(3)

Q2. "It felt like I could control the robotic body as if it was my own body"

strongly disagree	*disagree*	*somewhat disagree*	*neither agree nor disagree*	*somewhat agree*	agree	strongly agree
(-3)	*(-2)*	*(-1)*	*(0)*	(1)	(2)	(3)

Q3. "The movements of the robotic body were caused by my movements"

strongly disagree	*disagree*	*somewhat disagree*	*neither agree nor disagree*	*somewhat agree*	agree	strongly agree
(-3)	*(-2)*	*(-1)*	*(0)*	(1)	(2)	(3)

Q4. "I felt as if the robotic body was moving by itself"

strongly disagree	*disagree*	*somewhat disagree*	*neither agree nor disagree*	*somewhat agree*	agree	strongly agree
(-3)	*(-2)*	*(-1)*	*(0)*	(1)	(2)	(3)

Q5. "I felt as if I was located where the robot was"

strongly disagree	*disagree*	*somewhat disagree*	*neither agree nor disagree*	*somewhat agree*	agree	strongly agree
(-3)	*(-2)*	*(-1)*	*(0)*	(1)	(2)	(3)

Q6. "I felt as if the robotic body was matching the movements of my real body"

strongly disagree	*disagree*	*somewhat disagree*	*neither agree nor disagree*	*somewhat agree*	agree	strongly agree
(-3)	*(-2)*	*(-1)*	*(0)*	(1)	(2)	(3)

Q7. “I was able to adjust to the controls easily”

strongly disagree	*disagree*	*somewhat disagree*	*neither agree nor disagree*	*somewhat agree*	agree	strongly agree
(-3)	*(-2)*	*(-1)*	*(0)*	(1)	(2)	(3)

Q8. “I found the Heads-Up Display (HUD) useful in controlling the robot”

strongly disagree	*disagree*	*somewhat disagree*	*neither agree nor disagree*	*somewhat agree*	agree	strongly agree
(-3)	*(-2)*	*(-1)*	*(0)*	(1)	(2)	(3)

Q9. “Even though the depth perception of the camera was limited, I feel like I sufficiently adjusted to it”

strongly disagree	*disagree*	*somewhat disagree*	*neither agree nor disagree*	*somewhat agree*	agree	strongly agree
(-3)	*(-2)*	*(-1)*	*(0)*	(1)	(2)	(3)

Q10. “When asked to locate the Experimenter, I tried to find him as if I were in the robot’s place“

strongly disagree	*disagree*	*somewhat disagree*	*neither agree nor disagree*	*somewhat agree*	agree	strongly agree
(-3)	*(-2)*	*(-1)*	*(0)*	(1)	(2)	(3)

Q11. “Did you find it easy to perform the task left handed?”

strongly disagree	*disagree*	*somewhat disagree*	*neither agree nor disagree*	*somewhat agree*	agree	strongly agree
(-3)	*(-2)*	*(-1)*	*(0)*	(1)	(2)	(3)

Appendix K

Information Sheet

Participant Information Sheet

UCL Research Ethics Committee Approval ID Number: 4547/012

YOU WILL BE GIVEN A COPY OF THIS INFORMATION SHEET

Title of Study:
Robothespian Teleoperation
Department:
Computer Science
Name and Contact Details of the Researcher(s):
Nikolaos Tsikinis, nikolaos.tsikinis.18@ucl.ac.uk
Name and Contact Details of the Principal Researchers:
Anthony Steed, a.steed@ucl.ac.uk
David Swapp, d.swapp@ucl.ac.uk

1. **Invitation Paragraph**
   You are being invited to take part in a research project. Before you decide it is important for you to understand why the research is being done and what participation will involve. Please take time to read the following information carefully and discuss it with others if you wish. Ask us if there is anything that is not clear or if you would like more information. Take time to decide whether or not you wish to take part. If there is anything that is not clear or if you would like more information, please ask the experimenter. Thank you for reading this.

2. **What is the project's purpose?**
   The purpose of the study is to investigate the effectiveness of remotely operating a stationary bipedal humanoid robot (Robothespian) using our developed framework in Virtual Reality.

3. **Why have I been chosen?**
   We require that participants are able to see and hear unaided for the duration of the study, that participants do not consume alcohol within 6 hours before the start of the experiment and that participants are not sensitive to photosensitive epilepsy.

4. **Do I have to take part?**
   It is up to you to decide whether or not to take part. If you do decide to take part, you will be given this information sheet to keep and be asked to sign a consent form. You can withdraw at any time without giving a reason and without it affecting any benefits that you are entitled to. If you decide to withdraw you will be asked what you wish to happen to the data you have provided up that point.

5. **What will happen to me if I take part?**
   You will be asked to switch off mobile phones and to complete a demographics questionnaire. You will be shown a demonstration of the robot's capabilities and of the interface you will be seeing inside the program. You will then equip the Virtual Reality Headset and complete a series tasks by controlling the robot by using the controllers. The tests will include picking up objects and putting them inside designated boxes multiple times. Finally, you will be asked if you have any further comments about the experience and be paid £5 in cash. The whole experiment should take approximately 30 to 45 minutes.

6. **Will I be recorded and how will the recorded media be used?**

   Data such as tracking information from virtual reality equipment and your performance in the tests may be recorded. Any remarks made during the session might be written by the experimenters. Remarks that could lead to a potential identification of the participant will not be noted. It will not be possible to identify participants through tracking information, performance data or transcriptions. Tracking information, performance data and transcriptions may be used for analysis, demonstration and further research.

7. **What are the possible disadvantages and risks of taking part?**

   Users of virtual reality sometimes experience some degree of nausea. If at any time you wish to stop taking part in the study for this or any reason, please tell the researcher who will immediately end the experiment.

   Some users report 'flashbacks' and other side effects of using virtual reality equipment. Research suggests using virtual reality may cause short-term disturbances in vision.

   Virtual reality can be a trigger for photosensitive epilepsy.

8. **What are the possible benefits of taking part?**

   You will be paid £5 for your participation. Additionally, it is hoped that this work will improve understanding of what makes a person transfer ownership of his body to the controlled robot. The ultimate goal is to make virtual reality based teleoperation technology more successful.

9. **What if something goes wrong?**

   Should you wish to raise a complaint, please contact the principal researcher David Swapp using the contact details above. If your complaint is not handled to your satisfaction, you can contact the Chair of the UCL Research Ethics Committee at ethics@ucl.ac.uk.

10. **Will my taking part in this project be kept confidential?**

    All the information that we collect about you at any time will be kept strictly confidential. You will not be able to be identified in any ensuing reports or publications.

    Data such as tracking information from virtual reality equipment and task performance may be collected. It will not be possible to identify participants through this data.

    To facilitate removal of your data from the project should you wish to withdraw, a record matching your name and participant number will be made and stored securely. The record will be destroyed on or shortly after 25th July 2019. At this stage there will be no record of you having participated in the study, and your data will be anonymous.

11. **Limits to confidentiality**

    Please note that assurances on confidentiality will be strictly adhered to unless evidence of wrongdoing or potential harm is uncovered. In such cases the University may be obliged to contact relevant statutory bodies/agencies.

12. Use of Deception

Research designs often require that the full intent of the study not be explained prior to participation. Although we have described the general nature of the tasks that you will be asked to perform, the full intent of the study will not be explained to you until after the completion of the study, at which point you may withdraw your data if you wish.

13. What will happen to the results of the research project?

The results of the research will primarily be part of the students' final report and may or may not be part of a research paper in relevant journals or conferences in the future. Should you wish to obtain a copy of the published results, please contact the researchers using the contact details above. You will not be personally identified in any report or publication, and it will not be possible to personally identify participants from any data presented.

Anonymized data may be stored for subsequent research.

14. Data Protection Privacy Notice

Notice:

The data controller for this project will be University College London (UCL). The UCL Data Protection Office provides oversight of UCL activities involving the processing of personal data, and can be contacted at data-protection@ucl.ac.uk. UCL's Data Protection Officer is Lee Shailer and he can also be contacted at data-protection@ucl.ac.uk.

Your personal data will be processed for the purposes outlined in this notice. The legal basis that would be used to process your personal data will be the provision of your consent. You can provide your consent for the use of your personal data in this project by completing the consent form that has been provided to you.

Your personal data will be processed so long as it is required for the research project. If we are able to anonymize or pseudonymize the personal data you provide we will undertake this and will endeavor to minimize the processing of personal data wherever possible.

If you are concerned about how your personal data is being processed, please contact UCL in the first instance at data-protection@ucl.ac.uk. If you remain unsatisfied, you may wish to contact the Information Commissioner's Office (ICO). Contact details, and details of data subject rights, are available on the ICO website at: https://ico.org.uk/for-organisations/data-protection-reform/overview-of-the-gdpr/individuals-rights/

15. Who is organizing and funding the research?

The research is part of the Master of Research Virtual Reality program (TMRCOMSVRE01). The research is funded by University College London.

16. Contact for further information

David Swapp
Office: Malet Place Engineering Building, Ground Floor VR lab, WC1E 7JE.
Phone: +44 (0)203 5495 251 and +44 (0)20 7679 7211

Should you wish to take part, you will be given a copy of this information sheet and a signed copy of your consent form to keep. Thank you for reading this information sheet and for considering taking part in this research study.

Appendix L

Consent Form

PARTICIPANT CONSENT FORM

Please complete this form after you have read the Information Sheet and/or listened to an explanation about the research.

Title of Study:
Robothespian Teleoperation

Department:
Computer Science

Name and Contact Details of the Researcher(s):
Nikolaos Tsikinis, nikolaos.tsikinis.18@ucl.ac.uk

Name and Contact Details of the Principal Researchers:
Anthony Steed, a.steed@ucl.ac.uk
David Swapp, d.swapp@ucl.ac.uk

Name and Contact Details of the UCL Data Protection Officer:
Lee Shailer, data-protection@ucl.ac.uk

This study has been approved by the UCL Research Ethics Committee: Project ID number:
4547/012

Thank you for considering taking part in this research! The person organizing the research must explain the project to you before you agree to take part. If you have any questions arising from the Information Sheet or explanation already given to you, please ask the researcher before you decide whether to join in. You will be given a copy of this Consent Form to keep and refer to at any time.

I confirm that I understand that by ticking/initialing each box below I am consenting to this element of the study. I understand that it will be assumed that un-ticked/un-initialed boxes mean that I DO NOT consent to that part of the study. I understand that by not giving consent for any one element that I may be deemed ineligible for the study.

	Statement	Tick Box
1.	I confirm that I have read and understood the Information Sheet for the above study. I have had an opportunity to consider the information and what will be expected of me. I have also had the opportunity to ask questions which have been answered to my satisfaction.	
2.	I understand that I will be able to withdraw my data up to 25th August 2019.	
3.	I consent to the processing of my personal demographics information for the purposes explained to me. I understand that such information will be handled in accordance with all applicable data protection legislation.	
4.	I understand that all personal information will remain confidential and that all efforts will be made to ensure I cannot be identified except as required by law. I understand that my data gathered in this study will be stored anonymously and securely. It will not be possible to identify me in any publications.	
5.	I understand that my information may be subject to review by responsible individuals from the University (to include sponsors and funders) for monitoring and audit purposes.	

6.	I understand that my participation is voluntary and that I am free to withdraw at any time without giving a reason. I understand that if I decide to withdraw, any personal data I have provided up to that point will be deleted unless I agree otherwise.	
7.	I understand the potential risks of participating and the support that will be available to me should I become distressed during the course of the research.	
8.	I understand the direct/indirect benefits of participating.	
9.	I understand that the data will not be made available to any commercial organizations but is solely the responsibility of the researcher(s) undertaking this study.	
10.	I am aware that I will be paid £5 travel expenses. I understand that: a. I will still be entitled to this if I choose to withdraw, and b. beyond this I will not benefit financially from this study or from any outcome it may result in in the future.	
11.	I agree that my anonymized research data may be used by others for future research. No one will be able to identify you when this data is shared.	
12.	I understand that the information I have submitted will be published as a report and I wish to receive a copy of it. **Yes/No**	
13.	I hereby confirm that I understand the inclusion criteria as detailed in the Information Sheet and explained to me by the researcher.	
14.	This study requires that participants are able to see and hear unaided for the duration of the study, that participants did not consume alcohol within 6 hours of the start of the experiment and that participants are not sensitive to photosensitive epilepsy. I hereby confirm that: a. I understand the exclusion criteria as detailed here and as explained to me by the researcher; and b. I do not fall under the exclusion criteria.	
15.	I am aware of who I should contact if I wish to lodge a complaint.	
16.	I voluntarily agree to take part in this study.	
17.	I would be happy for the data I provide to be archived. I understand that other authenticated researchers will have access to my anonymized data.	

If you would like your contact details to be retained so that you can be contacted in the future by UCL researchers who would like to invite you to participate in follow up studies to this project, or in future studies of a similar nature, please tick the appropriate box below.

Yes, I would be happy to be contacted in this way	
No, I would not like to be contacted	

____________________________	__________________	____________________
Name of Participant	Date	Signature
____________________________	__________________	____________________
Researcher	Date	Signature

Bibliography

[1] "Telerobotics," [Date accessed: 31/5/2019]. [Online]. Available: https://en.wikipedia.org/wiki/Telerobotics

[2] R. Bushey, "Advancements in virtual reality device development," 2017, [Date accessed: 31/3/2019]. [Online]. Available: https://www.rdmag.com/article/2017/08/advancements-virtual-reality-device-development

[3] "Robothespian," [Date accessed: 31/3/2019]. [Online]. Available: https://www.engineeredarts.co.uk/robothespian/

[4] "Robonaut," [Date accessed: 31/3/2019]. [Online]. Available: https://robonaut.jsc.nasa.gov/R2

[5] C. Armel and V. Ramachandran, "Projecting sensations to external objects: Evidence from skin conductance response," *Proceedings. Biological sciences / The Royal Society*, vol. 270, pp. 1499–506, 08 2003.

[6] S. Nishio, T. Watanabe, K. Ogawa, and H. Ishiguro, "Body ownership transfer to teleoperated android robot," vol. 31, 10 2012, pp. 398–407.

[7] S. Kishore, "Robotic embodiment developing a system for and applications with full body ownership of a humanoid robot," 07 2016.

[8] A. Monferrer and D. Bonyuet, "Cooperative robot teleoperation through virtual reality interfaces." in *Journal of the Robotics Society of Japan.*, 2002. [Online]. Available: https://pdfs.semanticscholar.org/09e4/cbceb076ae85131cd2a4ea1c8c9f5d6a0515.pdf?_ga=2.98561944.646883423.1554124068-1079342843.1552242151

[9] T. Kot, P. Novak, and J. Babjak, *Application of Augmented Reality in Mobile Robot Teleoperation*, 03 2018, pp. 223–236.

[10] J. Chen, E. C. Haas, and M. Barnes, "Human performance issues and user interface design for teleoperated robots," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 37, pp. 1231 – 1245, 12 2007.

[11] P. Le Parc, E. Pardo, A. Touil, and J. Vareille, "Virtual reality to improve remote control in presence of delays," 10 2010, pp. 42 – 46.

[12] P. Reignier, F. Harrouet, S. Morvan, J. Tisseau, and T. Duval, "Arvi : A virtual reality multiagent platform," *Virtual Worlds 1998*, 01 1998.

[13] M. Gonzalez-Franco and T. Peck, "Avatar embodiment. towards a standardized questionnaire," *Frontiers in Robotics and AI*, vol. 5, p. 74, 06 2018.

[14] "Engineered arts," [Date accessed: 31/3/2019]. [Online]. Available: https://www.engineeredarts.co.uk

[15] "Stereoscopic vs. monoscopic 360 video for vr experiences," [Date accessed: 31/3/2019]. [Online]. Available: https://blog.vrdirect.com/stereo-vs-mono-360-video-for-vr

[16] "Ricoh theta v," [Date accessed: 31/3/2019]. [Online]. Available: https://theta360.com/en/about/theta/v.html

[17] "Htc vive pro," [Date accessed: 31/3/2019]. [Online]. Available: https://www.vive.com/us/product/vive-pro/

[18] "Steamvr plugin," [Date accessed: 31/8/2019]. [Online]. Available: https://assetstore.unity.com/packages/tools/integration/steamvr-plugin-32647

[19] "Bodyguards," [Date accessed: 31/8/2019]. [Online]. Available: https://assetstore.unity.com/packages/3d/characters/humanoids/bodyguards-31711

[20] "Cyborg character," [Date accessed: 31/8/2019]. [Online]. Available: https://assetstore.unity.com/packages/3d/characters/cyborg-character-112661

[21] "Space robot kyle," [Date accessed: 31/8/2019]. [Online]. Available: https://assetstore.unity.com/packages/3d/characters/robots/space-robot-kyle-4696

[22] "Final ik," [Date accessed: 31/8/2019]. [Online]. Available: https://assetstore.unity.com/packages/tools/animation/final-ik-14290

[23] "Quaternion," [Date accessed: 31/8/2019]. [Online]. Available: https://docs.unity3d.com/ScriptReference/Quaternion.html