POLITECNICO DI TORINO

Master Course in Computer Engineering

Master Thesis

Collaborative environment for shared virtual reality

Advisor
prof. Andrea Giuseppe Bottino

CoAdvisor:
prof. Alberto Raposo

Salvatore GURZÌ
matricola: 255212

To my family and all the ones that had believed in me

Abstract

Despite the cost reduction and the large availability of models on the market, the widespread diffusion of Head Mounted Displays (HMDs) in the consumer market is still far away. Thus, it often happens that Virtual Reality (VR) is a single-user experience that excludes bystanders. In contrast, cellphones and tablets are omnipresent, which makes them the perfect candidates to share the HMDs experiences.

We are proposing a CVE (Collaborative Virtual Environment) solution, where the users without an HMD could interact with the ones who have it using their HHDs (HandHeld Devices) .

We addressed the problem of creating an asymmetric collaborative solution by taking into account all the novels challenges associated with users that share different hardware.

To this end, we designed a shared application that enables to make 3D manipulations, while satisfying all the requirements needed to avoid the decrease of awareness due to the different views.

We conducted a user study and used the data collected to demonstrate that the application makes possible collaborations between HMDs and non-HMD users, giving our contribution alongside other research in the field.

Contents

3

List of Figures

Chapter 1

Introduction

This chapter has the purpose of introducing the work done, explaining the objective of the thesis and what it was developed, and of describing the document structure.

1.1   The objective

Each technology begins to be known when there is an interest in the market for it, in such a way its price starts to decrease and many people buy it. We believe that soon applications for both VR and AugmentedReality (AR) will be deeply developed. VR and AR are now used in educational environments (Hussein & Nätterdal, 2015), in psychotherapy (Diemer, Pauli, & Mühlberger, 2015), and in tourism (Guttentag, 2010)[1, 2, 3]. Less than two decades ago, VR and AR applications were feasible only behind closed doors with dedicated laboratories and powerful computers for testing and experimentation(Amir Dirin, Ari Alamäki & Jyrki Suomala, 2019)[4].

In May 2019, Ben Lang,a Road To VR [20] author, reported 1 million user connected on Steam for the first time and commented "After correcting for Steam's changing population, we find that May 2019 was the first month on record to see more than 1 million monthly-connected headsets on the platform. Year over year, monthly-connected headsets on Steam are up 80%.", which can be clearly observed in the Figure 1.1.

The prices of the HMDs are already decreasing and now more people can afford it, but it is still difficult to share the experience that you experiment with the non-HMD bystanders.

The thesis work intention is to make possible for phone/tablet user to interact with the HMD user through AR. The idea of this project is born in Tecgraf, Institute of Technical-Scientific Software Development of PUC-Rio (TecgrafPUC-Rio is a successful Brazilian model of self-sustained R,D&I (Research, Development and Innovation in different areas among which "Digital Interactivity" [5],) under the support of Alberto Raposo and Renato Cherullo that gave me the possibility to work on a thesis during my interchange in Rio de Janeiro.

The bounds of the real are broken from the virtuality, where users can interact using collaborative 3D manipulation that includes transformations, scales and rotations in order to increase the cooperation and to give different points of views.
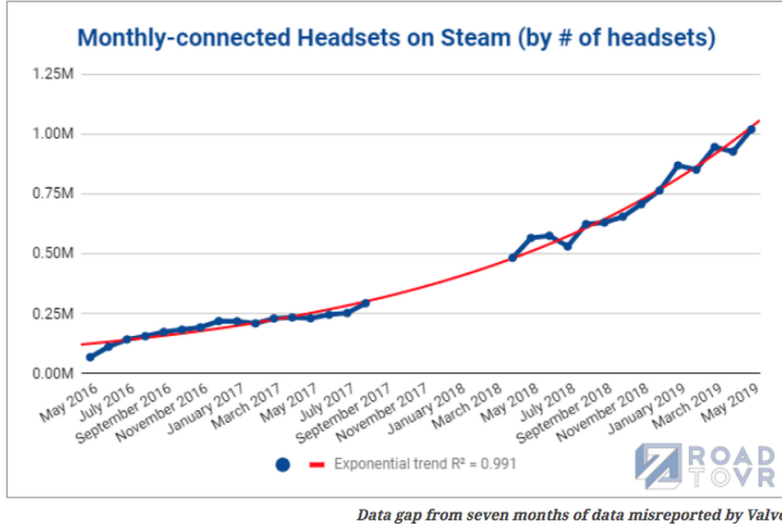
Figure 1.1: Graph showing the number or VR users connected from 2016 to may 2019, source: Road To Vear

Collaboration is possible only if every user can understand what the others are doing. Thus, as to maximize mutual understanding, we decided to include in the application a calibrating process in order to visualize the same position, rotation and scale of the virtual objects in both augmented and virtual world. This allows every user to see the exact manipulations of the objects made from the others. To this end we thought to use a real reference with an image positioned somewhere in the real world. The HMD user, before starting manipulating the objects existing in his dimension, should have positioned his controller on the reference and clicked a button *Figure 1.2*.

Following that, the phone user will need just to point at the reference and he will be able to see the objects in the same exact position, rotation and scale as seen from the HMD-user. To demonstrate that the different kind of users could really interact, we created three different construction tasks with an increasing level of difficulty. The implementation addressed all the typical difficulties that are met creating a VR/AR application, taking into account the relative solutions already provided and creating a particular one for our scopes. In such a context it is is important to maintain a shared code base between AR and VR so that the same code for the two kind of users is written just once in order to bring correction and improvements only in one place. Another extremely significant fact we considered is that collaboration, mainly based on communication, in such cases lacks of very important features (one user can't see the other because wearing HMD). Those must be substituted in some way adopting the appropriate solutions.

1.2 Structure of the thesis

In the next chapter we will discuss about the related work to this dissertation, in order to explain which is the current state of the art and which are the improvements added to the

Figure 1.2: Figure of how the headset controller should be positioned on the landmark to coordinate VR world with AR World

international knowledge with this work.

Then, in the Methodology chapter, we will focus on explaining which approach we used to answer the research question, how and why the application has been designed, in the Results,we will explain how the application was implemented, which user survey we adopted and our conclusions. Finally, in the Conclusions we will shortly summarize the work and indicate which could be the suggested further studies.

Chapter 2

State of Art

In this chapter we will address different topics related to the thesis' project in order to understand where this work is positioned within the current research framework and to show which are some of the possible solutions that were used to solve similar problem to ours. First we will talk about different interfaces that were adopted for 3D manipulation in AR and VR, then we will talk about the related work done in this area and to conclude we will show which can be the possible contributions of this work in such fields.

2.1 3D Manipulation Interfaces

Interfaces have always been important and are an expression of what the programmer think about the users, a special care is always needed in order to make possible for them to succeed in their tasks and to collaborate with each other.

2.1.1 AR interfaces

There are different devices for AR such as surface mapped projections, head-mounted see-through displays, and hand-held devices. The latter, usually smartphones and tablet, are ubiquitous nowadays and represents the main media through which we can access AR. Any device needs to interact with the objects that overlap in the real world, for this reason a well implemented interface is suitable, for such problem two different approaches are used:

- **Isomorphic** where hands actions are mapped in a 1:1 relationship to the virtual world, creating a more natural and intuitive way of manipulation

- **Non-isomorphic** exploits the multi-touch screens to identify different fingers gestures or the device's rotation and its position, that can be easily retrieved by the camera and the accelerometer, to move the object along with the HHD or a mixture of them.

The latter techniques, even if resulting less intuitive then the first, can break the human hands restrictions and allow an expanded way of interaction taking advantage of the unbounded virtual environment without losing considerable usability. These are widely adopted and differ in two main ways. The input is controlled directly or indirectly and

the position and the orientation: separately or together. Indirect manipulation can be achieved by a surrogate that can be in the same visualization screen or on another external apparatus. The latter are less used while the first are almost standard, nevertheless they are not adequate for small screens due to cluttering issues and fat finger problems [16].

Very adaptable results the DS3 manipulation technique [17] that allow a 6DOF (Degree of Freedom) interaction, we can in fact observe that a variation of it was adopted in [16] and in [15] leading them to obtain optimal usability scores during the user tests.

It can be classified as a separated technique with some characteristics from both direct and indirect interactions, allowing different manipulation through distinctive finger gestures. The fingers' position on the screen, on the object and their numbers allow to separate translation from rotation, for instance, two fingers on the object allows rotation while one direct finger and one indirect allow z-depth movement as can be seen from the resume technique table *Figure 2.1*

		Translation			Rotation		
	Mode	Tx	Ty	Tz	Rx	Ry	Rz
DS3	1d	○	○				
	1d + 1i	○	○	i	Z-Technique		
	≥ 2d				○	○	○
	≥ 2d + 1i			i	○	○	○

Figure 2.1: DS3 technique resume table, d/i: "direct/indirect" source: Géry Casiez

Multi-touch technique are often used together with device movement tracking, the first to use inertial HHD sensor to mimic natural movements for Mobile Augmented Reality (MAR) were Henrynson et al. [18] .

Marzo et al. [16] took into account previous works, saying device motion was the best for translating object while finger gesture performed the best in rotation, and decided to realize the transformation with the device movements and the rotation with the Arcball's metaphor [19].

Gustavo G. [15] instead implemented a variation of the 3DS technique with the possibility to switch in device motion modality through a GUI element on the right bottom border of the device. Objects are then manipulated applying to their coordinate system the modification made to the camera as shown in the *Figure 2.2*. No particular suggestion is given to the users, which can switch from a modality to another in base of the task they need to complete.

(a) Object attached   (b) Object translation   (c) Object rotation
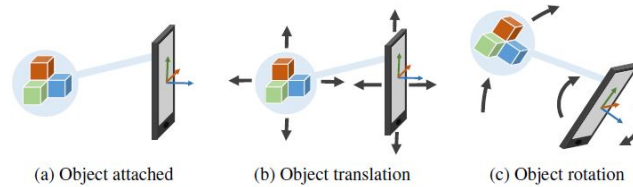
Figure 2.2: Gustavo device motion technique for 3D Manipulation. (a)the object is attached to camera transform (b) translate and (c) rotate with it

Using device movement and orientation as information results more intuitive and performs worse only in large rotation when techniques to avoid harms limits are not used. Nevertheless this allow to be free from finger gestures that could be difficult to use in small displays. We can conclude then saying that combining the two approaches: fingers gestures and device movement as stated in Asier et al. work [16] can be beneficial.

2.1.2 VR Interfaces

As discussed in the introduction (Ch.1), VR technologies are incredibly spreading and the number of VR headset sold increase month by month. For such reasons more studies are concentrating on improving the HMD interfaces that are becoming the standard for VR. Most of what is said about AR is also applicable for VR: we can distinguish in direct, indirect, isomorphic and non-isomorphic interfaces and they can break the human limits thanks to the virtual world.

The "gold standard" for VR is the virtual hand, the image of a controller or a hand is visualized in order to mimic the natural interaction (isomorphic 1:1) for translation and rotation within the arm's reach while two controllers can be used for scale. Unfortunately the 6DOF controllers used for such metaphor have some imperfections, they need to be tracked and don't work well on distances. The arm's reach limitation can be overcome using some buttons of the controllers, usually the *clutch*, in order to allow a complete 360° rotation. While distance problems can be beaten using non-isomorphic technique that gives that touch of "magic" to VR. *GO-GO technique*[23] uses the distance of the virtual hand beyond certain limits to allow the user to move a distant object with a projection of it, while behaving like a normal one when the hand is close to the body *Figure 2.3*.
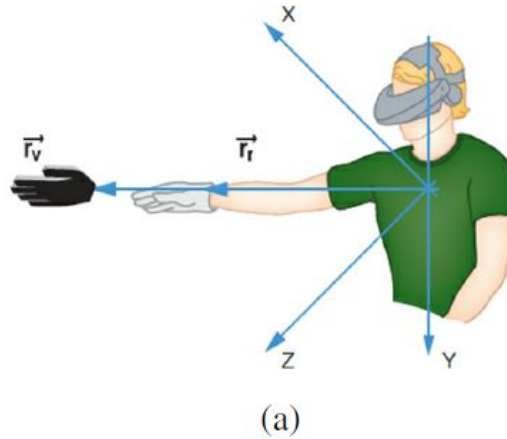
Figure 2.3: Go-Go technique, Source:Poupyrev et al. (1998)

Hybrid technique, such as a modification of the HOMER(Hand-centered Object Manipulation Extending Ray-Casting) [22], as stated from Gustavo [21], could reach the same time performance in collaborative environment and add benefits for co-manipulation applications: interaction on the same object with two virtual hands result complicated.

A ray cast from the virtual hand to an object allows to select it, then a different gesture allows manipulation and rotation with a mapping 1:1, although in some variants controller

buttons can block translation or rotation, similarly to what is done in the AR separated interfaces.

2.2   Related Work

The idea of this project was born, aside the interest of exploring the world of VR and AR, looking at a video on Youtube from NBC Left Field [6], where a girl explains which are the general wishes for the new year through sketches that she draws using the HTC Vive Controllers from the HMD of the same brand. Although the drawings we see in the video are clearly a photo-montage, the idea behind it was interesting therefore we decided to try to implement something similar, but using AR instead of the video editing technique.

According to the terminology defined by Milgram and Kishino in the article "*A taxonomy of mixed reality displays*" we can classify our work in the "*Virtual Continuum*" as "*Mixed Reality*"[7].

The virtual continuum *Figure* 2.4 is defined as an interval between reality and a complete virtual world defined as virtuality, while all that lies in between is defined as mixed reality. Other keywords connected to the work could be all that consider collabora-

Figure 2.4: Subdivision of the virtual continuum

tion in a virtual world, in an augmented one and the asymmetric co-located VR field.

The trendiest HMD devices at the moment still represent a pricey expense and occupy a lot of space, without considering the need of a powerful computer to make them work. The HTC Vive needs to be attached to the computer and two cameras to be around to track both the headset and the controllers. Meanwhile the Oculus Rift needs less space, you still need to position two cameras and to be close to the computer. Many efforts are being done to make VR mainstream, as making them more easy to use, removing the effort of positioning external tracking camera around the game area or selling HMD that doesn't need a valuable Computer to work with. Summing up it is reasonable that many people

started researching to understand how to create experiences where also the bystanders could participate without doing much effort and spending less money.

Many researches have been done about VR for phones with the relative investments, but with poor results. In 2016 Ed Baig wrote the article "Google Daydream fulfills wireless VR dreams" complaining for the lack of content of the Google Daydream *Figure* 2.5. "Here's the chief reality behind Google's $79 virtual-reality headset: It boils down to content [8] he said, the application Google Daydream that enabled him to use VR on the android phone Pixel, but still he wasn't happy with the lack of contents. He was right to be unhappy but for the wrong reason because, after 5 years, in 2019 Google announced the end of the platform.

Figure 2.5: Headset for Google Daydream

"We saw a lot of potential in smartphone VR," a Google spokesperson said. "...But over time we noticed some clear limitations constraining smartphone VR from being a viable long-term solution. Most notably, asking people to put their phone in a headset and lose access to the apps they use throughout the day causes immense friction"[9] . This lead to similar turn down destiny for other phone based VR technologies as Gear VR.

Other studies tried instead to focus on asymmetric ways to connect people.

Past researches in remote collaboration has found that sharing emotion can improve understanding and raise positive experience during collaboration, however, these systems were desktop based [10, 11]. In the study "*Sharing and Augmenting Emotion in Collaborative Mixed Reality*" [12], differently on what has been done in this work, they used an AR HMD and many MR devices as tablets to allow an artificial intelligence to share and augment emotion between different devices. The purpose of this work was to enable two people to work together remotely and if one of them feels anger or frustration the application recognize it from facial inputs like expression, eye-gaze, voice, heart-rate etc and augments the emotion and shows it to the partner using an avatar representing the collaborator. In such a way the workers can improve their collaboration recognizing more easily the emotions of their   partner   and   react   properly.

We know also about studies that instead focused on shared MR using co-located VR, two or more people interact between each other using different devices in an asymmetric way

with the help of tablets, television and projectors to augment the degree of involvement. This is the case of the work of Jan Gugenheimer et al. that created "ShareVR". The application enables co-located asymmetric interaction between users wearing an HMD and users without an HMD. ShareVR uses a tracked display as a window into the virtual world and a floor projection to visualize the virtual environment to all Non-HMD users. It enables collaborative experiences such as exploring a dungeon together, drawing, sports or solving puzzles as well as competitive experiences such as "Statues" or a swordfight. ShareVR facilitates a shared physical and virtual space, increasing the presence and enjoyment for both HMD and Non-HMD users[13].

In such studies we find involved the market of games, that is creating a lot of co-located VR applications that can be gathered in the field of "PartyVR". *Keep Talking and Nobody Explodes* and *Ruckus Ridge* are two example of shared, asymmetrical an co-located VR that had a lot of success despite adopting to different level of asymmetry. In the first procedural generated bombs must be detonated by the immersed player with the help from the non-immersed player equipped with a defuse manual. The player with the bomb can't see the manual and the other one has only his speech as a weapon to interact in the game. The latter is the first local-multiplayer mini-game tournament built for Oculus Rift and HTC Vive that pits 1 VR player against up to 3 friends in simultaneous combat. Ducking it out in a series of 1 vs. 3 and Free for All mini-games, players take turns in the VR headset while the others use regular controllers to play. Players score points by winning minigames and the player with the most points at the end of the tournament takes the victory![14] One work very similar to our is the one of Jerônimo Gustavo, who did studies in order to understand how 3D manipulation can be exploited to give the best results in collaborative environments. A collaborative environment provides more than one individual perspectives of the scene, which may also include shared and personal viewpoints. However, teamwork involves considerable negotiations and, as team members vary, team strategies and task-fulfillment processes also change. Modeling such human interactions raises novel concepts and interface design approaches when compared to those typically grounded in a single-user scenario[15]. The researcher set up three different scenarios:

- One AR-AR

- One VR-VR

- One VR-AR

For the third one, that is the one similar to our project, he adopted an analogous approach: he used Vuforia for what concerns the AR part, benefited of an Oculus Rift for the VR one and he made the two parties communicate through to the network using the library UNITY UNET. The work is about co-manipulation interactions and about finding what happens when people collaborate using different set of devices.

2.3   Contribution of the thesis

With our research in this field, we helped to give a clearer perception of what a developer of a collaborative application should keep in mind, explicating the possible problems that could be encountered and added valuable information to the global knowledge of the topic

through the results that we obtained, alongside our work with that of similar studies in the same sector as that of *Geronimo* [15] and *ShareVR*[13]. We can summarize the benefits that our work added to the world knowledge in this list:

1. Another model to support asymmetric inputs from different sources for 3D manipulations.

2. A hand held device interface for collaborative manipulations in augmented reality;

3. An interface for collaboration in immersive virtual reality;

4. A design of collaborative interactions for asymmetric mixed reality;

Chapter 3

Methodology

This chapter shows the approach we adopted, starting from the research questions to the development of the application, taking into account all the requirements and set of problems that one could face carrying out such experiments.

3.1 Questions of the research

As stated in the previous chapter the interest in VR and AR is growing rapidly not only in the research field, but in different areas. This leads to have a desire to get bigger audience of people acquiring the devices that make them accessible and to ask ourselves questions like the ones asked in this research.

With our studies we asked ourselves:

1. "Is it possible to share and enjoy the HMD user experience with the bystanders through handheld devices?"

2. "Is it also possible for such different users to collaborate?"

The method adopted to make those inquiries was to make an user study with an application that allows to collect qualitative and quantitative data. To this end we identified all the requirements that the application needed to our scope, afterwards we analyzed the annexed problems and finally we elaborated our solution.

3.2 Requirements of the Application

The nature of the studies makes possible to subdivide the requirements in two main fields:

- General requirements regarding the features of the application in the complex and of the two different parts(AR and VR)

- Requirements due to the "Mechanics of Collaboration"

The first group gathers all the requests and problems that concern both parts of the application such as:

- The choices of **different devices** to use, considering their hardware characteristics and their availability in the laboratory;

- The possibility for it to be run on different devices(**multi-device**) and on different platforms(**multi-platform**), without enormous effort for the user and the developer;

- A way to synchronize the different typologies of user in real time(**multi-player**), managing multi-connection and disconnection;

- The creation of **different tasks**, with increasing level of difficulty where the user can show that they are able to collaborate efficiently;

- Many **degrees of freedom** in order to make users able to manipulate virtual objects without considerable strain;

- The choice of a library for the support for the AR part, that would make the the visualization of the artifacts and the tracking of the position possible.

- The selection of a library for the VR part, which would allow to interact with different HMDs and to map the different controller keys to the adequate actions.

- The choice of appropriate **metaphors to manipulate** objects for both parts, considering the once already discussed in Chapter 2

While the second group gathers, as suggested from the item list, the features desirable in a groupware software. To collaborate doesn't mean to complete a task together (task-work) but to work as a team, which can lead to different issues that belong to two groups: those born from social and organizational factors and those resulting from inadequate support for the basic actions common in many work environments: "the mechanics of collaboration" [25]. While for the first there is not much we can do, for the second we need to consider how to address each of them:

- **Explicit communication**: Users should be able to exchange information: talking, writing or using gestural communication.

- **Implicit communication**: Group members should be able to understand what the other is doing through his movements or from the object that is being manipulated.

- **Coordination of action**: People should be able to collaborate without conflicts, using turns while manipulating artifacts or completing task following particular rules.

- **Planning**: being able to subdivide works in actions, plotting or reserve part of the workplace.

- **Monitoring**:The application must provide features to show who is currently in the work space to monitor his/her actions

- **Assistance**:Provide mutual support when an action is easier for one or another or when the other is in possession of more instruments or information.

- **Protection**:In collaborative environment someone can ruin your work so you have to be able to keep track of it and be able to protect it.

Figure 3.1: Oculus Rift

Figure 3.2: HTC Vive Pro

Figure 3.3: G. Hololens

3.3 Our solution

Selecting one device instead of another, while using the same application can lead to really different user experience [26]. Every choice made to fulfill the requirements should be made wisely and zealously, thus we decided to address the different problems considering not only which was better for our application, but also the constrains due to the environment and the time we had.

3.3.1 Multi-device

As said in the Chapter 1, we were inspired from doing this research through a video on youtube, where a girl wears an *HTC Vive Pro* Figure 3.2 which we had in the laboratory and results to be one of the best HMD in the market.

This particular headset has two comfortable controller which we could have used for the interaction and the calibration and a good system of tracking based on the camera in the front and the ones placed around.

In the Brazilian laboratory we also had the *Oculus Rift* Figure 3.1 and the *Google Hololens* Figure 3.3; even though the first is more commercial and would have helped for a possible spread of the application on a market it is less performing and also less comfortable to wear than the Vive; the second is a mixed reality headset, which basically means that we could have seen the same environment of the user with the handheld device while our studies focus on sharing only the same virtual objects.

For the non-HMD users we thought to target android modern tablets with a display in the range of 8.9", because we considered them to be of a comfortable size to observe the AR artifacts and to realize fingers gestures and we had one with such characteristics in the laboratory: the a *Nexus 9*. Nonetheless, thanks to the adaptive interface, the application works well also in standard devices.

3.3.2 Multi-platform

Our application basically needed to run on *Windows 10*, *android* and on the *Htc Vive* , but we wanted to support also *Linux*, *iOS* and the *Oculus Rift*. To this end we built our software on *Unity 3D* "The world's leading real-time 3D development platform", which we had already used and that allows to switch build platform very easily: "Create once, deploy anywhere".[27]

This software also grants the integration with many libraries for AR and VR, which was exactly what we were looking for. To create object in AR we could have used *ARCore* and

ARKit, two famous libraries in this field, that could have enabled us to avoid the adoption of a target, a better tracking position and plane detection but they have a trade-off: to be quite young and still supported only from the more recent apparatus. *Vuforia*, instead, is a robust technology that, with their leading computer vision, enables precise and powerful AR experience through many devices and platforms based mainly on target recognition [28] and compatible with a large range of devices. For us, the perfect candidate: we needed a landmark to create a point that would have been the origin for both the realities, and we wanted to allow as much users as possible to join the HMD experience through their devices.

In the VR field we considered two mainly diffused libraries:

1. *OpenVR* an application programming interface (API) that enables programming for VR hardware without the specific knowledge of it developed by *Valve*[1].[29]

2. *Open Source Virtual Reality* (OSVR) basically like *OpenVR* but open source and developed by *Razer*[2] and *Sensic*[3]

Unity supports both with the *OSVR Plugin* for OSVR and the *SteamVR Plugin* for OpenVR, we choose to use the second because created for the Htc Vive, easily supports the use of different HMDs and guarantees various tutorial and guides on the internet. Indeed thanks to the introduction of a document "the action manifest", which gives a greater level of abstraction and enables to separate the device used from the written code, the same actions mapped to the input keys of the Htc Vive controller can be mapped on those of different nature through really understandable interface *Figure 3.4*.
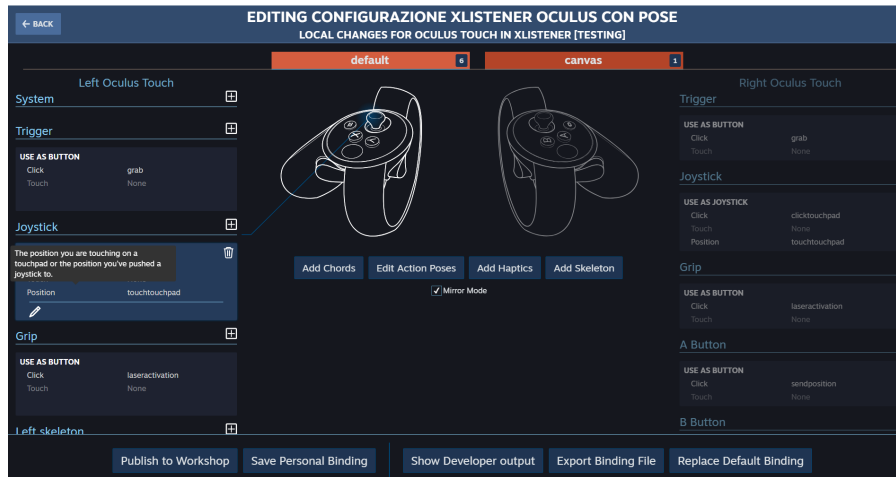

Figure 3.4: SteamVR Interface to associate controller inputs to actions

[1]An American video game developer, publisher, and digital distribution company

[2]A global gaming hardware manufacturing company

[3]Leader in the production of Virtual Reality Displays

3.3.3   Multi-players

The possible networking architecture solutions for a multiplayer application are two: peer-to-peer (P2P) or client-server; both have their advantages and disadvantages, but considering that we have to use a computer to run the headset application and that implementing a client-server application is way easier than doing a P2P, we decided to use the second strategy.

Unity offers two famous plugins for the network framework: *UNet*, which is being deprecated for the reason that it wasn't meeting anymore the requirements for modern multiplayer games [30]and *Photon Unity Networking* (PUN), which is tight to the application and ease to use with various demos inside the package [31]. Nevertheless both could have been good candidates for our solution, we wanted to have an infrastructure that we could use even if one day we would have decided to change Unity for another developing software, so we adopted *SignalR*. A software library of Microsoft *.NET*, a free open source cross platform developer platform [32], which enables to receive real time high-frequency information for maps, games and web pages [33] adopting the best protocol available for the communication.

3.3.4   Collaborative and Multi-task

If we look in a dictionary for the meaning of "Collaborate" on a dictionary, we can find that many defines it as "Work Jointly" which leads to a straightforward definition of a groupware: "A computer application that allows its users to collaborate with each other". Nonetheless we can't define activities like "chess game" not collaborative, in fact ,as Winograd[4] says, in some cases "Collaboration" more than a concept is a "state of mind" [34].

We wanted to create a solution that made people feel that "state of mind", hence the range of task that we could considered was restricted to those activities that would have made us reach such moods. Firstly we thought about creating something that could help people to build things: a task where people create a "Paper Bridge" *Figure 3.5*, that is a kind of structure that works without the use of any kind of paste.We abandoned the idea because this would have taken too much time for people to complete it.

Figure 3.5: A no-glue bridge made of paper

---

[4]Professor of computer science at Stanford University in America

Therefore we found an exercise that could have enriched the experience of seeing artifacts in 3D shapes, short to complete, but not too elementary to prevent or stop people from cooperating: "The 3D Docking Task". The docking task is widely adopted in the evaluation of manipulations interfaces [35] and consists in transforming a virtual object in one of its transparent copy with the same rotation, scale and translation. One task wouldn't have been enough to really test the application, thus we decided to create three different ones with an increasing level of difficulty based on a greater number of objects to manipulate and operations to complete the exercise.

To enable the transforming of the artifacts in AR we adopted a variation of the 3DS technique, that has told in the section 2.1.1, was used successfully from many researchers. While for what concern the headset, we used the hand metaphor considering the indirect and non-isomorphic technique not necessary for this task: the user is close to his work place and an the surrogate of a hand makes the interaction more natural.

The interaction with the virtual object was designed to give the same degree of freedom to both participants, allowing balanced participation, positive interconnection and individual importance. More exactly we provided 7 - DOF: 3 - DOF in the translation, 3 - DOF for the rotation and 1 - DOF for the scale.

The application was thought to have its user to be co-located, this means that explicit communication and monitoring, can happen in the more natural way possible through speech and sight. Nonetheless the user immersed in the VR can't see the others, so many of the features that normally helps "feeling" the social presence are missing. To overcome the issue and to support the "mechanics of collaboration" we inserted avatars, an outline to the objects when they are selected and played sounds during the interactions. The first ones help the HMD user to see the other participant and the HHD user to "feel" that the counterpart is also in the game, it supports implicit communication, coordination of action, monitoring (e.g one of the participant can retrieve information from the position and rotation of the avatar); whereas the outline helps monitoring and implicit communication: showing who is in possession of an artifact, planning and assisting: for instance, if the user sees me using an object he will try to move the remainders or try to correct my mistakes once I ended the manipulation and protection: red outline means that the object is being manipulated from the other player and I can't interfere until he has completed the action; At the same time the sounds, such as the slam one when the artifacts hit something, support monitoring what is happening and assistance(e.g. If I hear many smash sounds probably the other user is having difficulty to position the artifacts and needs help).

To enhance the creation of a strategy to achieve a task and to make unambiguous the meaning of the outline and what the users should do in each of them, we added some hints that are further discussed in the next chapter.

Chapter 4

Results

After dealing with the requirements and the issues about the creation of the application, we will discuss about the results obtained from those argumentations and about the user study thought to prove that collaboration between HMDs and HHDs is possible. The implementation we obtained is not the result of a single discussion, but something that is born also through some iterations of the process. Those interactions made us fix some of the problems which came out after the first trials (needed to record the data for the study), that otherwise would have compromised it.

4.1 Implementation

The program was developed to be used in a room, with one HMD user and one or more HHDs players in order to make them collaborate and share the same experience through a mixed reality environment and with the help of a physical marker.Firstly, we will discuss firstly what the system does, how is divided, how each part of it works, the instrument used. Secondly, we will talk about the experience realized during the study.

The application allows different kind of users to manipulate virtual dynamic objects via different metaphors of interaction based on the platform where they use it, showing the same environment if not for few static objects that are peculiar of the platform chosen. The HHD works like a window in the users world, showing AR objects which perfectly overlap the VR ones thanks to actions of calibration done at the beginning of the experience *Figure 4.1*. A client-server system records all the information about the users and the objects transform and keeps everyone up-to-date, allowing any other possible user to join the experience in any moment.

The system relies on two programs: the one that is in charge of showing the information accordingly to the platform where is being run, which we called `XListener`, and the one that manages, stores and keeps the information updated, named `WebApplication`.

XListener

It is an application completely developed in *C#* through *Unity 3D* that exploits the conditional compilation to use some scripts instead of others. Additionally the program enables

Figure 4.1: Figure showing how the VR user sees the other one manipulating an object, which perfectly overlaps in the two worlds

to visualize and manipulate object in both android and Hmd devices. This is possible thanks to the different "Scene" that are loaded once the program is run. A "Scene" is a peculiar feature of *Unity 3D*, a preset of objects and scripts that manages particular views or framework *Figure 4.2*.
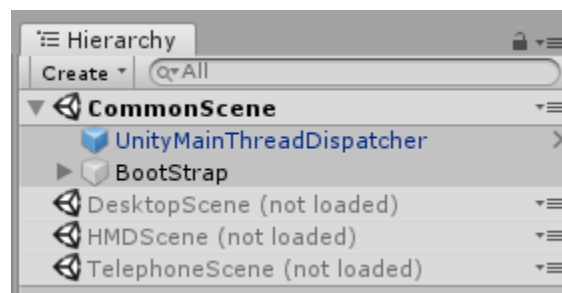
Figure 4.2: Screen of unity editor showing the "Scenes" created

We created three different ones:

- **The Common Scene:** this preset contains the core of the application which is a `ClientManager` script, it enables the start of the connection with the server and loads the particular scene depending on the platform where the application is run.

  A class, the `HubConnection`, is exploited to register to different asynchronous server calls and to invoke them in order to exchange information between the other users.

- **The Android Scene:** it is charged whenever in the *Unity IDE* the development platform or the device used is *Android.*

This scene exploits a peculiar *Unity* Plugin, **LeanTouch**, which was widely adopted to create the different finger gestures that enables translation, rotation and scale of the objects. The plugin scripts were modified in order to realize an interaction similar to the *DS3 technique* described in Chapter 2. For translation there are two modalities: one enables to move the objects up and down, while the second enables to move them on the xz local plane while locking their y coordinates.

To change modality the user has to double tap the screen with one finger. The screen will then show a text at the bottom of the screen and the mutation of a dynamic icon on the upper left corner to notify the modality transition *Figure 4.3.*

(a) The object can be moved only on the "XZ" plane

(b) Objects can be moved only up and down

Figure 4.3: Icons and texts used to alert the user of the movement modality

Whereas one single tap on the object will select it and one on the screen will deselect it, to translate the selected artifact the user will have to drag the finger through the screen, that will result in a movement in the xz or xy plane according to the modality currently in use.

In order to translate the virtual artifacts, the 2D inputs gesture where changed into a 3D outputs coordinates using a function that takes into account the position of the

camera in the world(the Z coordinate) to restore the object correct position in the mixed reality environment.

Dealing with this translation can be challenging, in fact when a user moves the world XZ coordinates don't, performing movement of the object must be made thinking about its local coordinates that stays the same. In this, the Unity `Transform` class helps keeping track of the global and local coordinates of the object with the properties `position` and `localPosition`.

Additionally if the user touches the screen s/he can perform different manipulations according to the distance and the gesture used *Figure 4.4*:

– A**Roll**: if the two fingers are at determinate positive threshold distance and they perform a pinch and after a twist. The roll, that usually is a rotation around the local x axis, is performed around an axis that has the opposite direction of the forward vector of the camera, which makes the action more intuitive.

– A **Uniform Scale**: if the gesture is a zoom.

– A **Yaw** or a **Pitch**: if the two fingers are really close and move in parallel upwards or swipe lateral. Specifically: a positive rotation around the camera Y axis if the swipe is on the right or a negative one if is on the left and a positive rotation on the opposite direction of the camera Z axis if the swipe is upwards or a negative one if it is downwards.
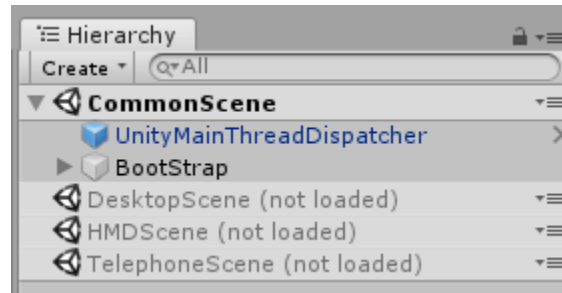
Figure 4.4: Gestures

One big challenge in doing such interface was due to the *Gimbal Lock* problem, it consists in the loss of one degree of freedom when one of the gimbals overlaps another, blocking the system in an undesirable condition where there is no method to accommodate a rotation around one axis [36]. Practically this translates into an undesired rotation around a different axis then the one chosen when performing yaw, pitch or roll (e.g. When a user tries to perform a roll, a twist is performed). To overcome this issue, those rotations can be performed using the Unity class `Quaternion` and in particularly the function `AngleAxis` that internally solves the *Gimbal Lock* problems.

For what concerns the virtual objects visualization, as said in the Chapter 3, it was implemented thanks to the *Unity* plugin *Vuforia*. The user should hold his device and frame the landmark, this will allow him to see the virtual objects, a wood table that is meant to improve the understanding of the depth of the different artifacts and a red button, which functionalities will be explained later *Figure 4.5a*.

(a) Table seen from the HMD user

(b) Table seen from the HHD user

Figure 4.5: View of the application tables, where the scenes take place

Additionally thanks to the Vuforia extended tracking feature, that can be enabled activating the tracking of the device pose from the configuration panel, it is possible to keep visualizing the virtual objects also if you frame something slightly distant from the marker. This one is an image of many little stones with a Cartesian plane indicating the xz axes print on the top, the particular picture allows its recognition and the Cartesian arrows facilitate the calibration process with the HMD *Figure 4.6*.
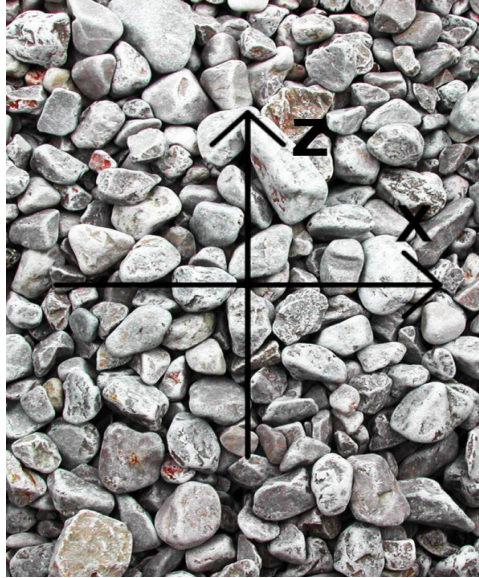
Figure 4.6: Landmark used for Vuforia

This kind of visualization influences the dimension of the objects, that means that every time the HHD user needs to receive information about the artifacts or to send them the program has to filter them and apply a scale factor, that has to be calculated accordingly with the dimension of the target.

- **The HMD Scene**: it is the scene that is loaded when the platform is the computer and an HMD is available. The environment is a little bit different from the Android

29

Scene. It is possible to see the HTC Vive controllers, a tiled blue floor, a different wood table (even if similar) and the red button. In order to improve the understanding of the virtual room depth, the floor has a tiled texture, while the wooden table has also the steams, that contrary to the AR version, that doesn't have them, it doesn't overlap any existing table and resulting more natural *Figure 4.5b*.

The scope of this part is to enable the HMD user to interact, visualize the virtual world and to calibrate it with the one of the AR users. The visualization and the interaction in the virtual environment is possible thanks to a particular Unity plugin `SteamVR`, that exploits the `OpenVR` library.

From the Window panel in Unity is possible to manage different controller configurations in order to associate their key inputs with certain actions. The status of the button can be checked through coordinates values (e.g. where the key input was touched) or thanks to a boolean value (e.g. true if the key was pressed, false otherwise) that are returned from a particular action. Thus when the user presses the trigger button *Figure 4.7* and the top of his controller is close to an object, he selects it.Afterwards, if he keeps holding the trigger, every movement associated with the controller is applied also to the object, as if the controller was a real hand. Instead if, s/he clicks on the touch-pad *Figure 4.7*, it evenly scales the object: making it larger, if s/he had clicked the upper part of it or smaller, if s/he had clicked the lower one. Pressing the lateral key, the grip, the reference system to which all objects are related in the world moves, in such a way the user can calibrate it to the one of the counterpart in AR positioning the controller parallel at the direction of the z axis, drawn on the *Vuforia* marker.
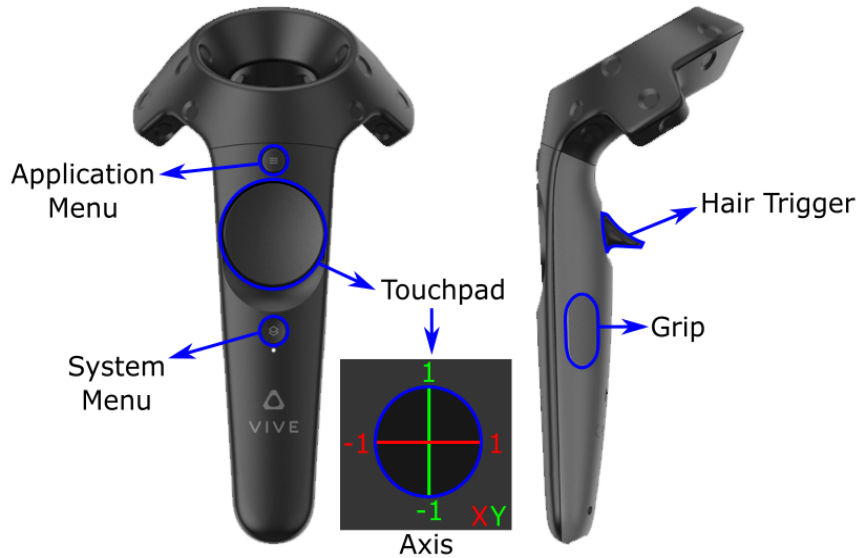

Figure 4.7: HTC vive controller with the map of all the buttons

WebApplication

It is the implementation of the server and has the purpose of making the users communicate: it stores the information of the 3D objects, loads the experience environment requested, keeps track of the users currently active and writes the logs of the application. The scope of this part is to make users have a synchronous vision of what is happening in the environment: seeing the same preset of artifacts, hearing the sound of an object colliding with something and observing what the others are doing.

The exchange of the data between the clients is made across a system of asynchronous incoming and outgoing calls, that travels with the Websocket transport protocol where is available or to older protocols where is not. This is possible thanks to the `SignalR` library, part of the ASP.NET framework, already described in the Chapter 3. To wrap the information in classes, for better handle them, it was created a package called `SharedClass`, that has the name suggests, it is shared with the program `XListener`. Therefore the features of each object are stored in a class common to both programs, `Primitive`, that is stored in a dictionary `PrimitiveHolder` based on the object's name. Whenever an artifacts is created or a user connects, a field in the dictionary is added, thus when other clients want to join the experience they can have up-to-date data. Similarly every time an object is manipulated, a user is moving it or a new object preset is loaded, the information in the classes nominated above are updated and the clients alerted *Figure 4.8*.

Figure 4.8: Diagram Description how our messages are handled by the server

To write the log were used ".csv" file using the library `.CsvHelper` recording:

- Client and session information

- Object information about who moved what, for how long and which kind of transform s/he did applied.

The Experience

As mentioned in Chapter 3 the experience consists in three different "docking task" of increasing level of difficulty. The users have to manipulate the objects in order to make

them match their transparent copies and then press the red button, upon the table, to visualize what their accuracy in positioning was and proceed to the next task *Figure 4.9*. To improve the immersion, the presence and to give support to the "mechanics of

Figure 4.9: Screenshot taken from the HHD application version showing the canvas with the instructions to complete the task

collaboration", cited in Chapter 3, many features were added to the application:

- It was implemented a "**Collision System**", whenever a manipulated object collides with something a slam sound is played in both the games, the first available position not overlapping with anything it is calculated and the object is moved to that location. This helps the users to implicit communicate, to monitor each other actions, to give support in case one of them finds difficult to position a object and to increase immersion and presence because makes everything seem more real. Unity already has a system of collision based on a feature called `Rigidbody`, but it was not adopted to make the application run lighter and faster. We preferred to use the Physics Engine of Unity only in part: there was no need to calculate gravity forces, thus the functions used to calculate the nearest possible position when an object is overlapping with another are called only in particular situations. The artifact has to be selected and then every 30 Frame Per Second (FPS), a function , `Physics.OverlapSphereNonAlloc`, casts a sphere: having as center, the object center and as a radius, the distance between the center and the further mesh point from it, checking if there are objects in its neighborhood. In the positive case another function checks if the object is colliding and if it is, the program plays a sound and calls the server to alert the other user of the bump, in such a way the sound can be heard also from the other clients.

- A "**Ownership policy**" was created in order to make users understand when they selected an object and to protect the work they were doing: one artifact can be manipulated by one person at time. When the user tries to select the object his outline becomes green, meanwhile it is red when selected from someone else *Figure 4.10* .

  Every user is assigned a unique integer id by the server of an unique integer Id that it is used to manage the ownership of an object: when the user tries to move an artifact a check on its Id is done, if it is "-1" its outline becomes green and the program sends
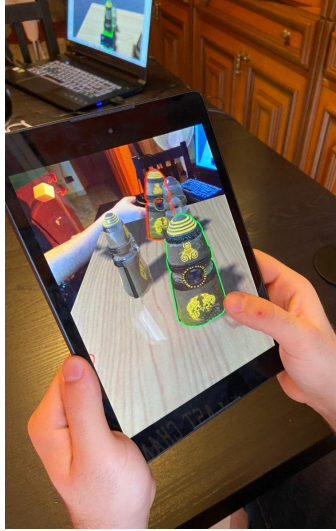
Figure 4.10: Picture of a player holding the HHD and showing the outline of its current selected object(in green) and the one selected from its partner(in red)

a message to the server to alert the other clients that the object owner is changed; otherwise the manipulation movements have no effects on it. This was possible using a *Unity Plugin* called `QuickOutline` that enables to activate/deactivate the outline of an object and select its color by script.

- A "**Tracking User Position System**" that increments the social presence showing the user movements through the use of avatars: an HMD headset for the HMD users *Figure 4.11b* and a tablet for the HHD ones *Figure 4.11a*. The avatars are created as all the other objects but only in the other users views, the field `Shape` in the class `Primitive` helps the application to know which avatar to create: a tablet if it is "android" and an HMD if it is "hmd".

  The color of the avatar is handled thanks to the Id that is assigned at the beginning, its value in fact influences the color, whereas its position is updated by a particular script `SyncPosition` that monitors the position of the camera and alerts other users of the changes when these exceed a certain threshold.

The user can access the different experiences through a "menu", a canvas which features were made more pleasant to the eye using the Unity Plugin `TextMeshPro`. Depending on the device used, the user can touch, if he is using an HHD, or he can point, through a laser beam that starts from the controller and reaches the menu window, and press the grip button, if he is using an HMD, one of the canvas buttons to select a task to perform. Whenever one of the four button are pressed, the canvas is closed, a preset of object is loaded and, in the HMD version the laser beam is turned off. The possible choises are :

- **Level One**:This scene has the purpose to make the users practice and approach with the application. It has to be completed only translating three "magic stones" identical models in their transparent duplicates *Figure 4.12*.

(a) HHD's avatar seen from the HMD user
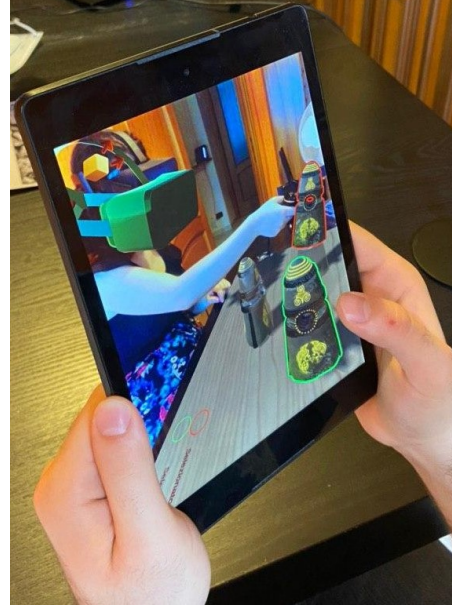
(b) HMD's avatar seen from the HHD user

Figure 4.11: Pictures of how the avatar are seen from the different devices

Figure 4.12: Screenshot taken from the HMD application version, showing the scene of the first task

- **Level Two**: This level is thought to make the users also use also the rotation transform and to improve their skills in the manipulation of the objects. They have to build an "ancient Japanese temple" *Figure 4.13* made of eleven elements: five to be translated and six to be translated and rotated.

  At first we decided to not give any suggestions to the users on how to build the

34
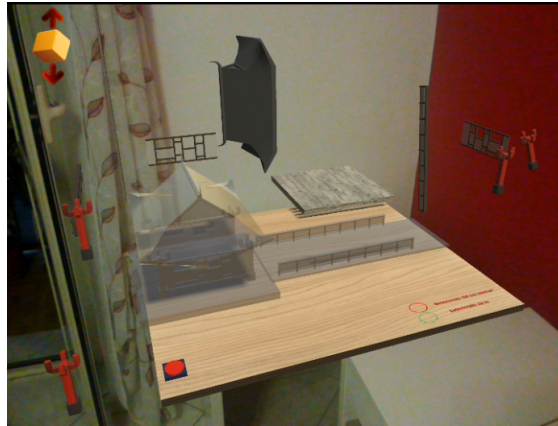
Figure 4.13: Screenshot taken from the HDD application version, showing the scene of the second task

construction, but then we saw they found difficult to start positioning the model in the correct order (e.g they were starting from the roof rather than from the foundations), thus we added some suggestions. We divided the construction in steps: first they needed to put the infrastructure, then the railing, the column, the walls and at last the roof.

For each step, every time a user selects an element that doesn't belong to the current phase the application shows a message on top of it "I would rather select another element in this moment" *Figure 4.14* and shows nothing if he s/he selects the correct ones.

Figure 4.14: Screenshot taken from the HDD application version, showing the suggestion message translated in Italian

We avoided blocking the user both from selecting some elements (this way, in fact, it would have been too easy for the group to understand what to do) and to not block

them from moving an object in order to have more action space.

- **Third Level**: This final task is slightly different from the others and was thought to encourage people to collaborate not only on how they need to position the objects but also on the objects placement. The idea for this challenge is born thinking about a similar one in a famous museum in Rio De Janaiero, *The Museum of Tomorrow*, in one room there was a game where players had a set of food and they had to create a balanced meal dragging them in a virtual menu. Similarly, in this challenge the users have to manipulate(rotate, translate and scale) different kinds of food and to collocate them in their transparent twins, but differently from the other tasks, they also need to place them in the correct box considering their higher nutritional content *Figure 4.15*.

Figure 4.15: Screenshot taken from the HDD application version, showing the third level scenario

> The elements present were put to be divided equally in the carbohydrate and protein trunk and their composition for 100g of product is described in the table 4.1 build taking the information from [40].
>
> For what concern the manipulation actions to be performed: the cabbage and the egg have to be translated and scaled; the fish, the cheese, the onion, the potato and the brown bread translated and rotated.

Additionally to the main menu, there were adopted other canvases: two spawned at the beginning of first and third level to give more information on what to do in the tasks; one, spawned at the end of task two and one, to show the total percentage of accuracy reached in the object positioning and to allow to continue the experience through a button and another, at the end of everything, to show the accuracy and which elements were positioned in the correct place.

The accuracy, reached from the users, is calculated by an event that is fired from the touch of the red button, which triggers the operations done from the script `CalculateDockDifference`. Every transparent object has one of it attached, it calculates the difference in position, scale and rotation of it with respect to another object, found in its surrounding. The script looks

Food	Carbs for 100g	Proteins for 100g
Cabbage	83.9%	12.7%
Fish	0-7.9%	10-22%
Cheese	1.3%	25%
Brown Bread	48.5%	7.5%
Onion	84%	15%
Egg	1.1%	13%
Potato	17%	2%

Table 4.1: Protein and carbohydrate percentage values per 100 grams of food

for the closer object that has the same "tag": a particular Unity objects property that allows to distinguish between them and then do its calculus. Depending on the level, it skips some of the comparisons (e.g in the first one ignores rotation and scale), because not necessary for the task scope. The results obtained are then normalized in a value between (0,1) using this formula:

$$(Diff - Max)/(Min - Max)$$

"Diff" stays for the specific difference in position, scale or rotation; while "Min" and "Max" are the peculiar threshold of tolerance mentioned in the table 4.2.

Action	Max	Min
Rotation	20°	5°
Scale	0.5 m	0.05 m
Translation	1 mm	22 mm

Table 4.2: Min and Max thresholds to calculate users accuracy in manipulation

A value close to "Min" is a very good result and translates into one point, while a value close to "Max" or bigger it is considered as a complete mistake and translate into zero points. The values obtained through this process are then normalized again, for the second and third level; with a "Min" of zero for both and a "Max" of two, for the second and three for the third. The results calculated are then sent to a core script, `GetTotalScoreGeneral`, where an average, based on the level elements, is done and then transformed in percentage.

4.2   User Study

This scope section is to explain which were our investigation fields, what kind of people were involved during the tests, which were our choices made for the protocol adopted and how the study was conducted.

4.2.1 The investigation fields

As mentioned before in *Chapter 3*, our investigation fields were the methods to bring people closer to the virtual reality world, allowing HMD users bystanders to interact with them through an HHD device.
We asked ourselves if it would have been possible for the last, not only to share the same participation, but also to collaborate with the first. The supposition was that it would have been less easy to perform certain actions on the tablet than with the HMD, nonetheless they would have been able to succeed helping each others in the collaborative tasks in the potential applications.

For such reason for one week we asked a group of people, with the age range between 20-30 and few experiences with AR and VR, to perform three different collaborative tasks.

The objective was to collect the most significant data out of which were our possibilities, recording qualitative data through questionnaires and quantitative thanks to the application logs.

We adopted two separate, but with specular questions, fact-finding surveys for the HHD and HMD users, three famous qualitative questionnaires: the Functionality, the Consistency and the Flexibility modules of the *VR Usability Evaluation*, (**VRUSE**) [38]; *The System Usability Scale*, (**SUS**) [37] and the *Social Presence Module* of *The Game Experience Questionnaire*, (**GEQ**) [39].
Instead, with regard to quantitative data, during each session were recorded:

- The accuracy and the time within each team completed the tasks

- The food put in the wrong trunks

- For each object of each level, for each specific test session every time it was manipulated, the actions performed (translation, rotation and scale), the ownership time and the device used to interact (HMD or HHD) were recorded.

More information about the questionnaires, why and what they specifically inspect will be examined in the specific session, while now we will concentrate on how the test session was performed.

4.2.2 Adopted procedures

For the experimentation we adopted the google tablet *Nexus 9*, an *Predator Triton 500* and an *Oculus Rift*. We didn't use the *HTC Vive Pro* because it was not available in the laboratory when we needed it, therefore we also extended the application also to the headset mentioned above.

Due to the specific historical period in which we found ourselves, after the period of lock-down due to the Covid-19 virus spreading, the number of tests were limited to a restrict number. Moreover we adopted specific measures in order to perform them safely.

Figure 4.16: Thesis author while sanitizing the Oculus Rift used for the experimentation

Each device adopted was sanitized before each experiment, adopting non-alcoholic antibacterial wipes while holding it with plastic gloves *Figure 4.16*.

In each test session, which lasted around 30/40 minutes, the users had to go through three different docking task positioning virtual opaque artifacts in their transparent copies. The testers we chose were people that never tried the application before and didn't know which was its purpose, but were advised that they would have contribute to a research and that their actions were recorded by the application.

At the first test iteration we did to not implement any tutorial, but for many people especially the ones that never experienced VR or AR those bunch of information given all at once were too much to remember; therefore, we introduced it on the second iteration we introduced it. Thus after the players completed a fact-finding survey, before starting the experience, they did a tutorial which introduced them to the specific application manipulation metaphors of the HHD or HMD device.
Each session concluded with the giving of the different questionnaires: the **GEQ** Social Presence Module, the **SUS** and the Functionality, the Consistency and the Flexibility modules of the **VRUSE**.

4.2.3 Questionnaires

The purpose of the study is to validate the thesis that people with HHD can collaborate with the HMD users, through quantitative and qualitative data. Defining what is to truly collaborate is not a simple issue and what we are trying to show here is that people were able to: help each others and "feel" they had been of any support.

There are different way of doing that, our approach was to adopt a questionnaire to inspect the "feeling" part with questions about the social presence and to analyze the data we collected. For what concerns the first part we adopted the GEQ social presence module which is commonly used to analyze the social presence through question about feelings the player had during the experimentation and it is widely used:in fact two of the study cited in *Chapter 2* adopted it([13] and [15]).

Revealing that the average efficiency rate obtained in the docking task was high, that people were both handling different objects and performing various actions and a good score in the social presence module would have clearly given credit to our hypothesis, but it wouldn't have been enough. We lacked to demonstrate that the information we were storing was from a stable application easy to use, we needed to prove its usability was at least on the average.

A "quick and dirty" questionnaire to define the usability broadly, to collocate our artifact in a general background in order to know if it was the general range of the acceptable usability applications is the SUS one, thus we adopted it.

The context where an application is used is very important and it is extremely bond to the usability of it, so to define the latter you always need to be really specific [37]. We needed also something specific that worked for our background too, thus firstly we thought to adopt two different solutions the VRUSE for VR and the *A Usability Scale for Handheld Augmented Reality* (HARUS)[1] for AR, but then there would have been the need to adapt the different questionnaires to have the exact same scopes.

For such reasons and to have an evaluation of the application interface we ended up using only the VRUSE and not even all of its parts, because some of them covered the same topic of the SUS and other were not of our concern. We took the Flexibility, Consistency and Functionality modules, discarding other ones like the Simulation Fidelity or the Error Correction/Handling Robustness, because we were not concerned by those investigations.

Every questionnaire was translated in Italian and they can be found in the Appendix:

- Fact-finding questionnaires *Appendix A*

- GEQ - Social Presence Module questionnaire *Appendix B*

- VRUSE- Funtctionality, Consistency and Flexibility modules *Appendix D*

- SUS *Appendix C*

---

[1]Questionnaire that considers perceptual and ergonomic issues found in HAR

4.3   Results' Discussion

Due to the particular period, because of the diffusion of the virus covid-19, we didn't have the chance to gather a big amount of data and from various people as suggested from usability studies [38]. Nonetheless we tried to made our best, with the instruments we had and to make suppositions with the information collected.

4.3.1   General information about participants

Two distinct questionnaires, with slightly different questions, were given to the two kind of users. They had to answer to four questions, three of which were measured with the Likert scale: a point of one means they totally disagree from what it's said, while a point of five that they absolutely agree. The answers can be seen in the *Figures 4.17* for HHD users and in the *Figures 4.18* for VR users.
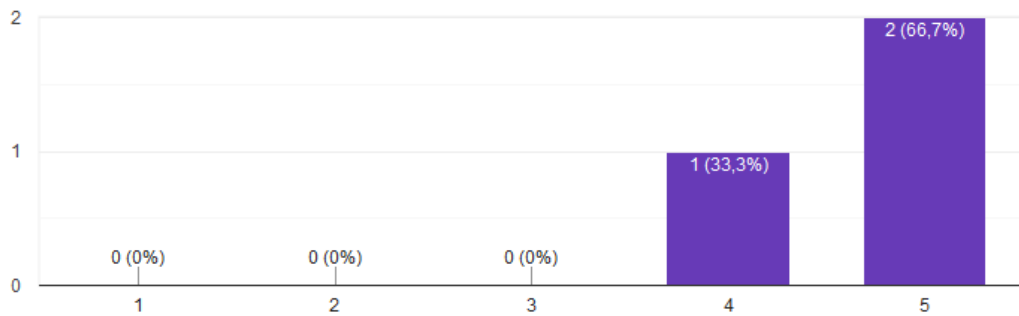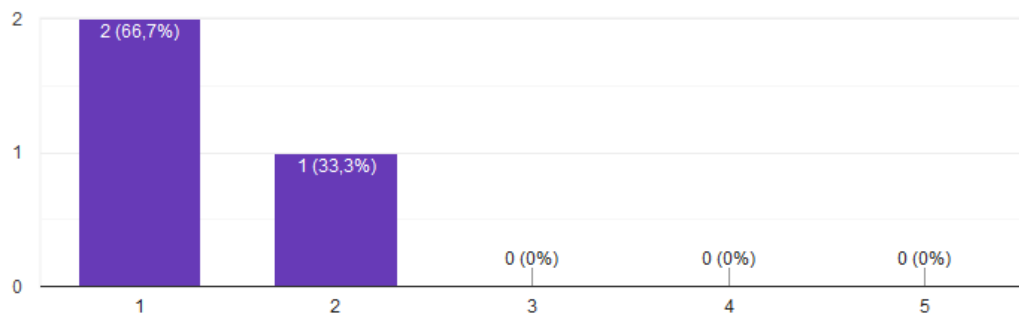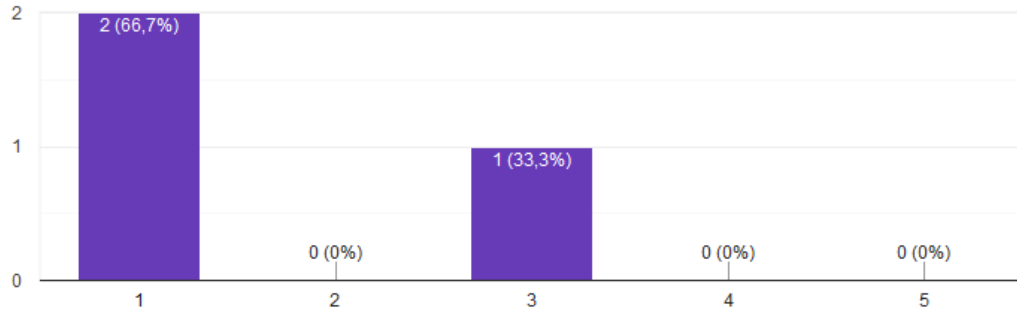

Figure 4.17: Part of the HHD user testers fact-finding questionnaire questions

While the data from the other questions were kept separated, the information about the age range and the question 4) were merged for the sake of representation and because

2. I have a good knowledge of immersive virtual reality

3 risposte

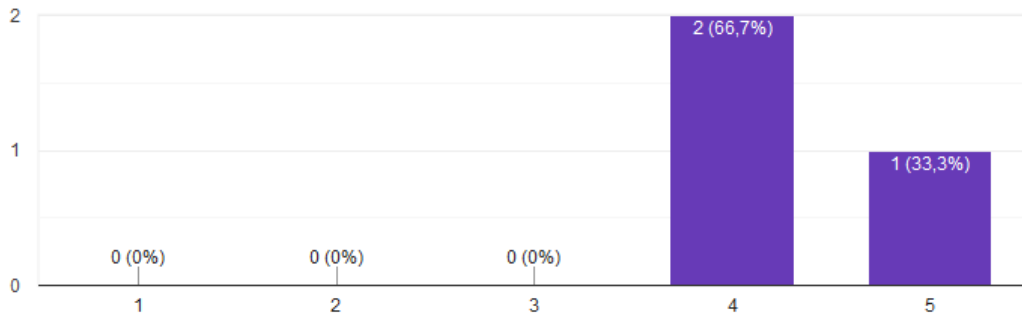3. I am interested in virtual reality

3 risposte

Figure 4.18: Part of the HMD user testers fact-finding questionnaire questions

we believed it made more sense to show them together *Figures 4.19* .

From the data can be observed that our testers were young people 20-30 years, they had used groupware applications but they had poor knowledge about VR immersive and AR applications. They had used tablets before and were interested in VR.

4.3.2 Evaluation of the usability

As discussed before, for the usability evaluation we used two distinct questionnaires: one more general and for broad applications, the SUS and one relative to our context, the VRUSE.

Evaluation of the usability made with the SUS The SUS questionnaire is made with a Likert Scale, alternating positive to negative answers to keep high the user attentions, and it covers various topics of the system usability such as: the need for training, support, and complexity [37].
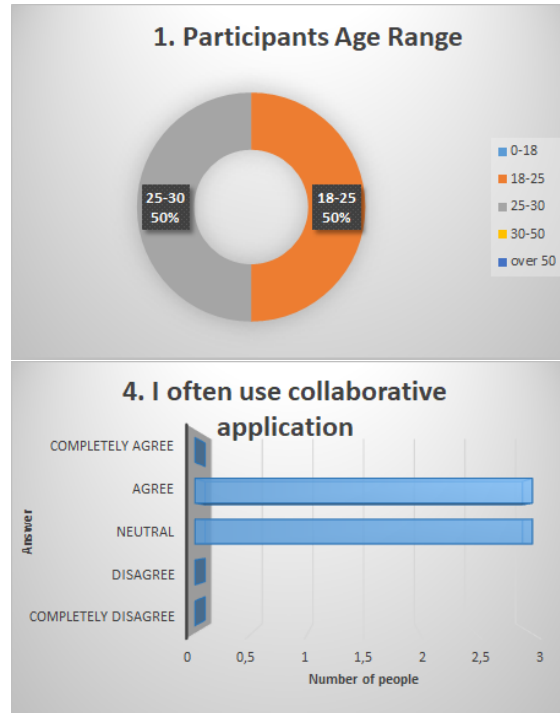
Figure 4.19: HHD and HMD testers information about age and general knowledge of groupware

After calculating the score as described in the study, we found out to be on 80/100, which is over the broad threshold of a good application of 68/100.

Evaluation of the usability made with the VRUSE

The data collected thanks to the VRUSE evaluation demonstrate us that, in overall, the testers thought that the interface provides a level of functionality (control) they expect in order to complete a task, that the operation of the system are consistent with theirs understanding and conventions and that they were not constrained by the system and able to use it in a flexible manner as can be seen clearly from the *Figure 4.20* and from *Figure 4.21, Figure 4.22 and 4.23* where the negative questions were marked with a "(i)".

Considering the general responses to the two questionnaires we can say that the system is easy to use and it is flexible, consistent and functional.

4.3.3   Social presence assessment

The information gathered about the social presence is also in the format of the Likert Scale and has to be read and discussed separating the pieces of information in question groups, because they inspect different aspects of the human behavior: empathy feelings, behavioral involvement and negative feelings.

The more relevant group for the study is the behavioral involvement *Figure 4.25* ,

7. Overall I would rate the VR system in terms of functionality as:

6 risposte

55. Overall I would rate the consistency of the system as:

6 risposte

61. Overall I would rate the flexibility provided by the system as:
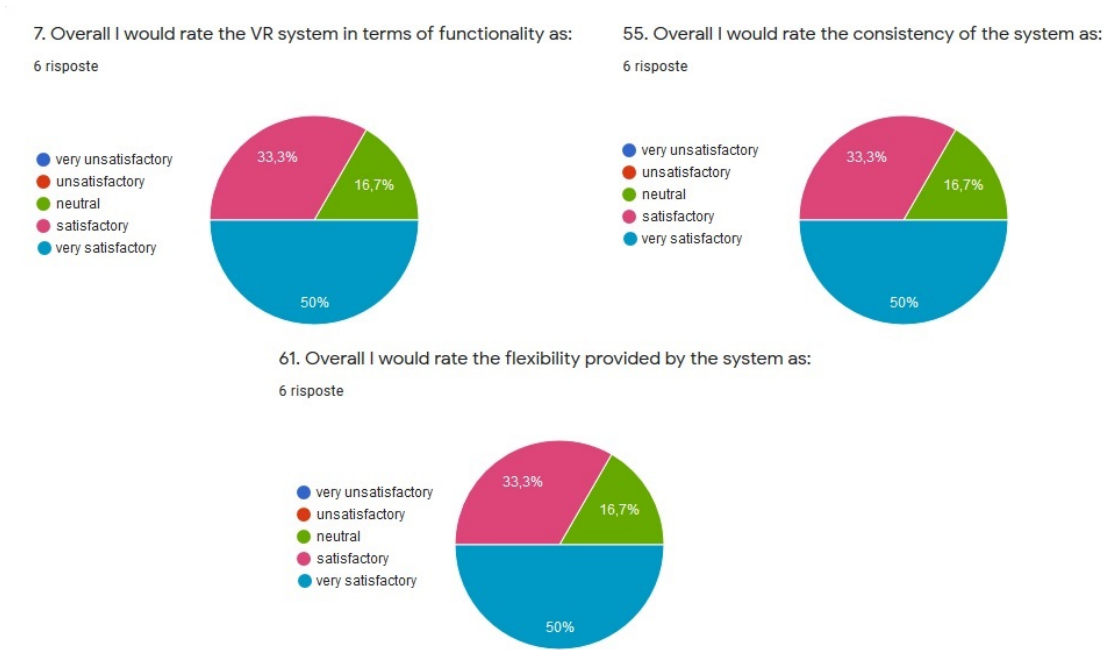
6 risposte

Figure 4.20: Pie chart graphs showing the answers for the "overall questions" of the functionality, consistency and flexibility modules from VRUSE
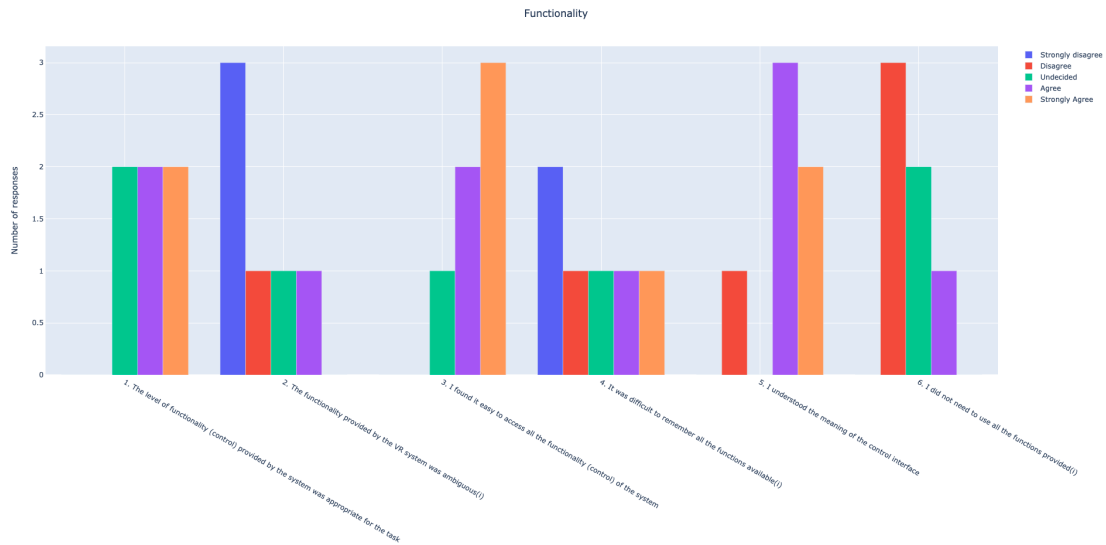
Figure 4.21: Answer to the question from the Functionality Module of the VRUSE

as can be noticed: there was a general agreement about each others actions were correlated(q.2,q.3) and about how what the other did affected what they did(q.14), which is positive, meaning that they were not working on their own ; while they were not secure of influencing the other(s), which is less positive, but tell us to improve the acknowledgment
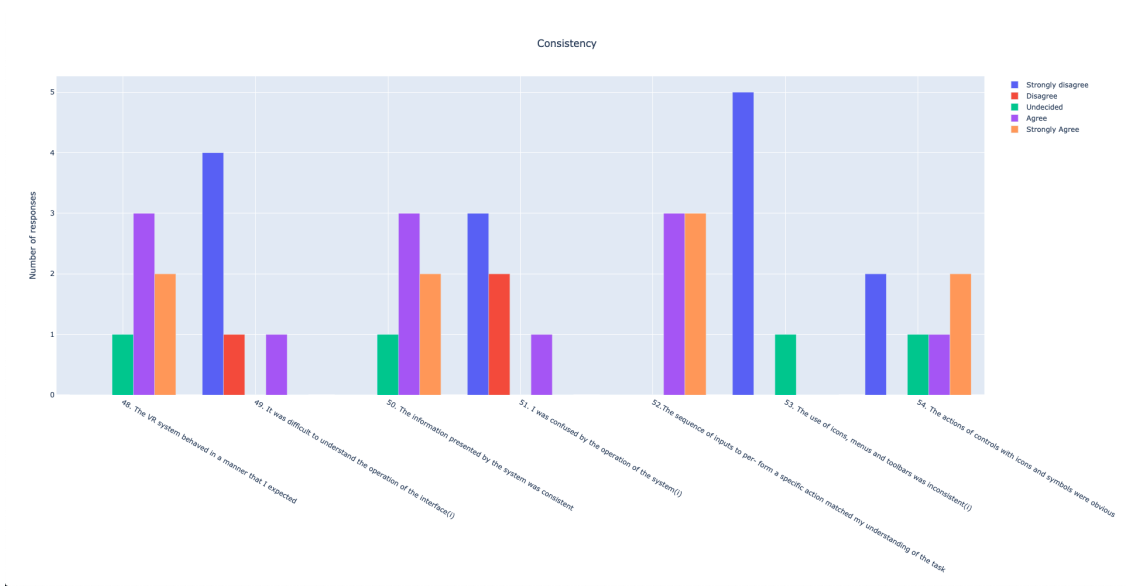
Figure 4.22: Answer to the question from the Consistency Module of the VRUSE
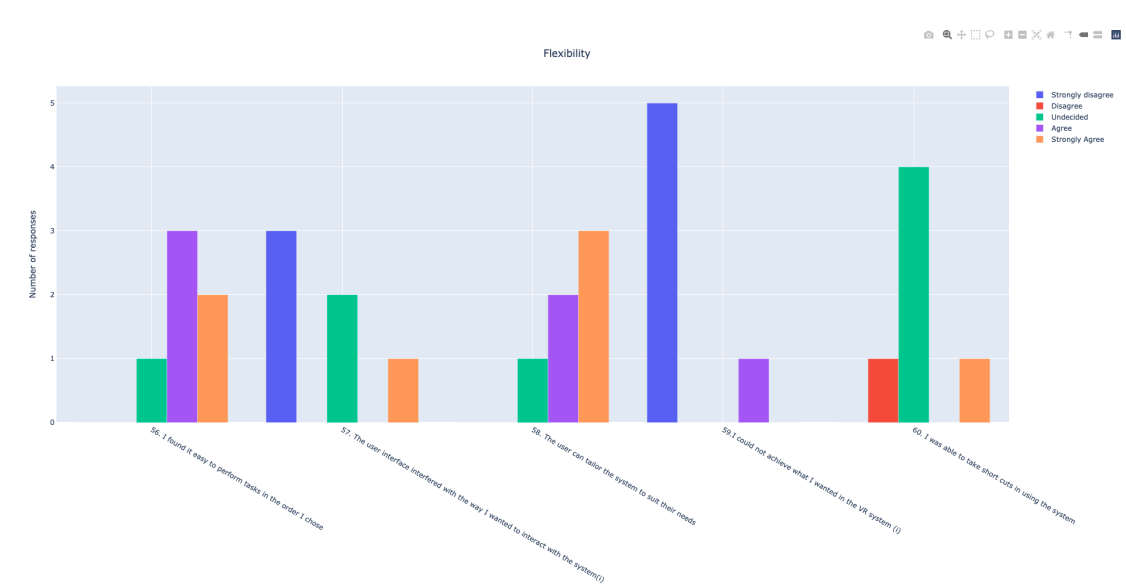
Figure 4.23: Answer to the question from the Flexibility Module of the VRUSE

on how their own actions influenced the other(s).

For what concerns the feelings, which are also very important, the data tell us that for the positive ones: the testers felt enjoyable to be with the other(s) (q.8) , happy (q.9,q.10) and they had admiration for their partner(q.13), while they were not really confident on declaring to feel connected (q.4) . Instead, for the negative ones we can observe: the average of them strongly agreed about not feeling jealous, revengeful or schadenfreude

(q.7,q.16,q.17), which leads to say that they were not frustrated for the collaboration; while we can't notice the same data arrangement for what concerns their moods (q.11,q.12).

Adding up, it can be affirmed that the users behaved and felt positively during the test sessions, which can be witnessed also by us, that had heard them talking and planning strategies politely, but also that some answers show us the possibility of improvement on "the connecting part".
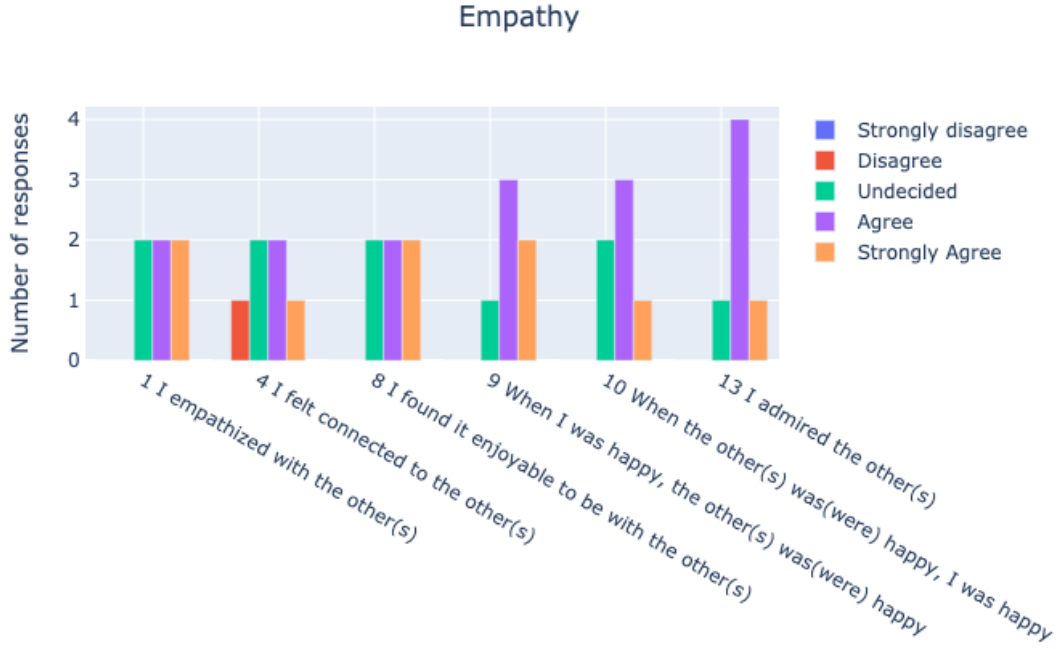

Figure 4.24: Answers to the questions about positive feelings from the GEQ - Social Presence Module

4.3.4 Evaluation of the quantitative data

In order to analyze the data recorded in the .csv[2] format, some pieces of the information were converted in *sql* by an online *CSV to SQL Converter* and then analyzed with the help of the *DB Browser for SQL*, while others where analyzed using *Microsoft Excel*.

DB Browser for SQL helps managing the database written in *MySql*, creating tables and Sql queries. We exploited it to join the data about the test sessions(device used, user id and session id) with the information about the total translations, rotations, scales and the sum of the overall time the player used to perform them, grouping the actions by the device adopted as can be seen from *Figure 4.27a* and *Figure 4.27b*. From those graphs we can see that the HMD users performed overall an higher number of operations in relation

---

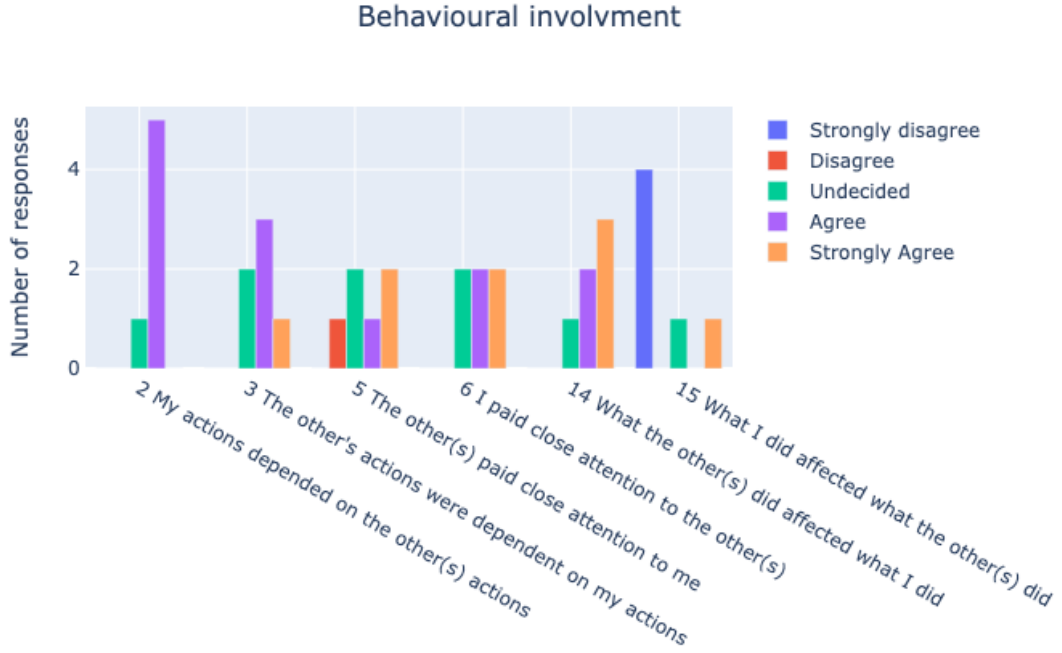[2]a format that separate the distinct values by commas

Figure 4.25: Answers to the questions about behavioral involvement from the GEQ - Social Presence Module

to the android users, while the time spent during the transformation is very similar: a total difference of 5 minutes. These results can be read, considering also what we experienced during the test sessions, as if the android user, s/he had more difficulty to manipulate the virtual artifacts due to the less intuitive interface respect to the HHD, which would explain why the time spent manipulating is higher. We can also observe that the disparity between the number of actions for rotation is the greater(as we expected), in fact, during the tests some of the android users manifested some troubles performing this operation.

Instead, for what concern the efficiency the users gained collaborating it can be noted from the *Figure 4.28* that all the tester gained satisfactory results, with scores all above the fifth percent, which means that more than the half of the objects were docked with the correct position, rotation and scale.

Thus, even if, the HHD users didn't manipulate as much as the HMD users, we can recognize, that the two kind of players were able to separate their work efficiently and to obtain a more than acceptable result and completed each task in a reasonable time *Figure 4.29* that would allow similar application to collaborate.

Figure 4.26: Answers to the questions about negative feelings from the GEQ - Social Presence Module
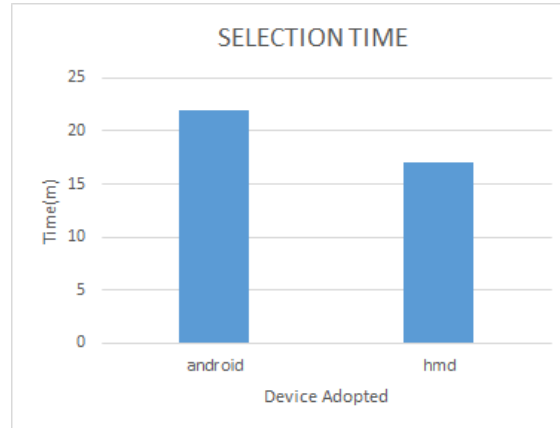
(a) Number of operations performed for translation, rotation and scale by each typology of device
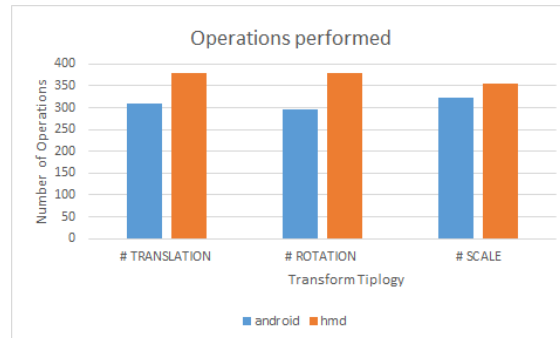
(b) Total amount of time spent manipulating various objects from the HMD and the android users

Figure 4.27: Information retrieved from the log files using DB Browser for SQL

Figure 4.28: Efficiency scores gained from the users for each level in the different test sessions

Figure 4.29: Average time needed by each group to complete the three tasks

Chapter 5
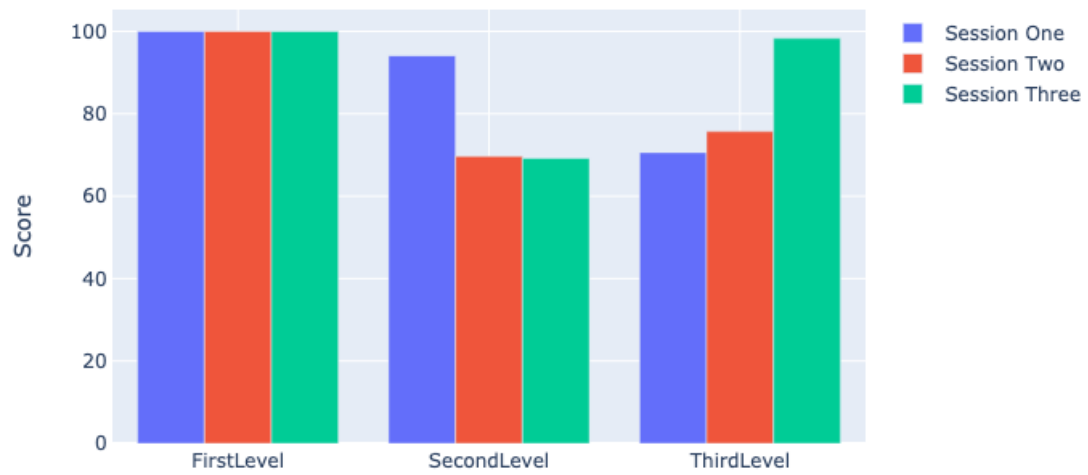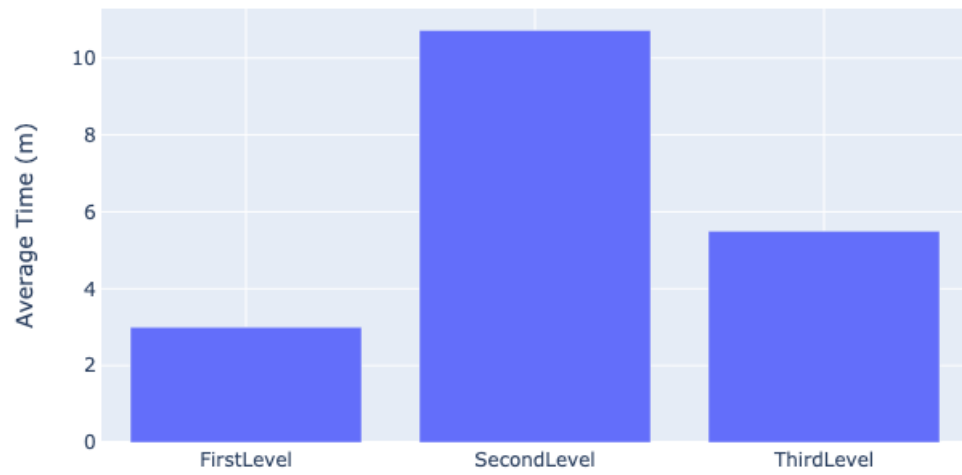
Conclusion

5.1 Summary

The thesis purpose was to investigate on how to bring more people closer to VR, allowing the bystanders of the HMD user to cooperate with him/her by HHD devices.
We asked ourselves if this would have been possible, created a collaborative application and made analysis taking care of qualitative and quantitative information. Those experiments lead us to confirm our hypothesis that HHD devices, such as tablets or cellphones, could be used to collaborate with the HMD users, even if, we found out with less agility.

With our studies we contributed to the researchers' knowledge:

- Adding our results to other similar researches, with information about collaborative virtual environment that concerns virtual and augmented reality

- Performing experiments about new combination of finger gestures that were not used before for manipulation

- Giving a possible implemented solution for similar systems

5.2 Further studies

The data collected from our research are not representative of all the population, which means that our statements should be fatherly investigated gathering more and differentiated data.
On this direction more experiments could be conducted:

- Allowing this application to work with the promising devices that works as HMD for AR. New opportunity of research, with the popularity of such MR interfaces, could include the exploration of the collaborative manipulations in MR scenarios.

- Improving the server implementation, making it run also on World Area Network

- Making experiments with bigger groups (eg. more than two people for each test session) and with different combination of devices

- Making progress about social presence, exploring others strategies to make the user feel more secure on how the others are feeling

Bibliography

[1] Hussein, M., & Nätterdal, C. (2015). *The Benefits of Virtual Reality in Education: A Comparison Study. University of Gothenburg*, Chalmers University of Technology. https://doi.org/10.1177/0011000003253155

[2] Diemer, J., Pauli, P., & Mühlberger, A. (2015). *Virtual Reality in Psychotherapy. In International Encyclopedia of the Social & Behavioral Sciences: Second Edition.* https://doi.org/10.1016/b978-0-08-097086-8.21070-2

[3] Guttentag, D. A. (2010). *Virtual reality: Applications and implications for tourism. Tourism Management.* https://doi.org/10.1016/j.tourman.2009.07.003

[4] Amir Dirin, Ari Alamäki & Jyrki Suomala, (2019). *Gender Differences in Perceptions of Conventional Video, Virtual Reality and Augmented Reality.* https://doi.org/10.3991/ijim.v13i06.10487

[5] Tecgraf, *Quem somos* https://www.tecgraf.puc-rio.br/publicaboutindex

[6] NBC Left Field, *Most New Year's Resolutions Fail, so Why Do We Make Them? | NBC Left Field* https://youtu.be/SNlUgMyg8R8

[7] Milgram and Kishino , *A taxonomyof mixed reality displays*

[8] Baig, Ed. *Google Daydream fulfills wireless VR dreams. USA Today, 14 Nov. 2016, p. 05B. Gale Academic Onefile,* https://link.gale.com/apps/doc/A470278517/AONE?u=puc_rjbr&sid=AONE&xid=d0720a17

[9] Chris Wiltz on October 28, 2019 *Op-Ed: Phone-based VR Is Dead, Long Live VR* https://www.designnews.com/electronics-test/op-ed-phone-based-vr-dead-long-live-vr/19541365361762#:~:targetText=It's%20the%20end%20of%20an,discontinuing%20its%20Daydream%20View%20headset.

[10] U. Eligio, S. Ainsworth, C. Crook, *Emotion understanding and performance during computer-supported collaboration,* Computers in Human Behavior, vol. 28, pp. 2046-2054, 2012.

[11] G. Molinari, G. Chanel, M. Betrancourt, T. Pun, C. Bozelle Giroud, *Emotion feedback during computer-mediated collaboration: Effects on self-reported emotions and perceived interaction,* 2013.

[12] Jonathon D. Hart*, Thammathip Piumsomboon, Gun Lee, Mark Billinghurst *Sharing and Augmenting Emotion in Collaborative Mixed Reality* School of Information Technology & Mathematical Sciences, University of South Australia https://www.researchgate.net/publication/332757691_Sharing_and_Augmenting_Emotion_in_Collaborative_Mixed_Reality

[13] Jan Gugenheimer, Evgeny Stemasov, Julian Frommel, Enrico Rukzio *ShareVR: Enabling Co-Located Experiences for Virtual Reality between HMD and Non-HMD Users* https://dl.acm.org/citation.cfm?id=3025683

[14] Foreignvr, *Ruckus Ridge VR Party* https://store.steampowered.com/search/?publisher=Foreignvr

[15] Jeronimo Gustavo Grandi, *Collaborative 3D Interactions and Their Application on Virtual, Augmented and Mixed Reality Interfaces* https://lume.ufrgs.br/handle/10183/187968?show=full

[16] Asier Marzo, Benoît Bossavit, Martin Hachet, *Combining multi-touch input and device movement for 3D manipulations in mobile augmented reality environments*, pp.14 https://dl.acm.org/doi/10.1145/2659766.2659775

[17] Martinet, A., Casiez, G. and Grisoni, L. *Integrality and separability of multitouch interaction techniques in 3D manipulation tasks.* IEEE Transactions on Visualization and Computer Graphics, Volume 18, Issue: 3 (2012). https://ieeexplore.ieee.org/abstract/document/5963662

[18] Henrysson, A.; Billinghurst, M.; Ollila, M. *Virtual object manipulation using a mobile phone.* In: Proceedings of the 2005 international conference on Augmented teleexistence - ICAT '05. New York, New York, USA: ACM Press, 2005. p. 164. ISBN 0473106574. http://portal.acm.org/citation.cfm?doid=1152399.1152430

[19] Shoemake, K. *ARCBALL: a user interface for specifying three-dimensional orientation using a mouse.* Graphics Interface (Vol. 92, pp. 151-156). (1992).

[20] *Analysis: Monthly-connected VR Headsets on Steam Pass 1 Million Milestone* https://www.roadtovr.com/monthly-connected-vr-headsets-steam-1-million-milestone

[21] Jeronimo Gustavo Grandi, *Collaborative 3D Interactions and Their Application on Virtual, Augmented and Mixed Reality Interfaces* pp.76 https://lume.ufrgs.br/handle/10183/187968?show=full

[22] Bowman, D. A.; Hodges, L. F. *An evaluation of techniques for grabbing and manipulating remote objects in immersive virtual environments.* Proceedings of the 1997 symposium on Interactive 3D graphics - SI3D '97, p. 35–ff., 1997. ISSN 03032647. http://portal.acm.org/citation.cfm?doid=253284.253301.

[23] Poupyrev, I. et al. *Egocentric Object Manipulation in Virtual Environments: Empirical Evaluation of Interaction Techniques.* Computer Graphics Forum, v. 17, n. 3, p. 41–52, 1998. ISSN 0167-7055. http://doi.wiley.com/10.1111/1467-8659.00252

[24] UnityTools, *CaptainHillman* https://github.com/CaptainHillman/UnityTools

[25] Potts Steves, M. et al. A Comparison of Usage Evaluation and Inspection Methods for Assessing Groupware Usability. pp.2 https://dl.acm.org/doi/10.1145/500286.500306

[26] Yücel Uzun, Real-Time Augmented Reality Collaboration with Different Device Types. pp.23 https://www.researchgate.net/publication/335890118_Real-Time_Augmented_Reality_Collaboration_with_Different_Device_Types

[27] Unity, https://unity.com/

[28] Vuforia Engine, https://engine.vuforia.com/engine

[29] OpenVR, https://github.com/ValveSoftware/openvr

[30] UNet Faq, https://support.unity3d.com/hc/en-us/articles/360001252086-UNet-Deprecation-FAQ

[31] Unity Photon, https://www.photonengine.com/pun

[32] .NET, https://dotnet.microsoft.com/

[33] ASP.NET SignalR, https://dotnet.microsoft.com/apps/aspnet/signalr

[34] Prasun Dewan, *Definition of a Collaborative Application*, http://www.cs.unc.edu/~dewan/290/s97/notes/intro/node2.html#SECTION00011000000000000000

[35] Jeronimo Gustavo Grandi, *Collaborative 3D Interactions and Their Application on Virtual, Augmented and Mixed Reality Interfaces* pp.57 https://lume.ufrgs.br/handle/10183/187968?show=full

[36] Wikipedia, *Gimbal lock*, https://en.wikipedia.org/wiki/Gimbal_lock

[37] John Brooke, *SUS - A quick and dirty usability scale* https://hell.meiert.org/core/pdf/sus.pdf

[38] Roy S. Kalawsky, *VRUSE - a computerised diagnostic tool: for usability evaluation of virtual/synthetic environment systems* pp.14, 16 https://hell.meiert.org/core/pdf/sus.pdf

[39] W.A. IJsselsteijn, Y.A.W. de Kort, K. Poels, *The Game Experience Questionnaire* pp.7 https://pure.tue.nl/ws/files/21666907/Game_Experience_Questionnaire_English.pdf

[40] Amparo Machado, *Scienza e Cultura Dell'Alimentazione*, vol.5 2015.

Appendix A

Fact-Finding Questionnaires

Questionnaires used to know how the set of testers was composed in order to correlate it to the other data collected

A.1   Fact-Finding AR questionnaire

A.2   Fact-Finding VR questionnaire

Appendix B

The Game Experience Questionnaire

Part of the GEQ - the social presence module used to evaluate the feelings of the users while approaching the application

- I empathized with the other(s)

- My actions depended on the other(s) actions

- The other's actions were dependent on my actions

- I felt connected to the other(s)

- The other(s) paid close attention to me

- I paid close attention to the other(s)

- I felt jealous about the other(s)

- I found it enjoyable to be with the other(s)

- When I was happy, the other(s) was(were) happy

- When the other(s) was(were) happy, I was happy

- When the other(s) was(were) happy, I was happy

- I influenced the mood of the other(s)

- I was influenced by the other(s) moods

- I admired the other(s)

- What the other(s) did affected what I did

- What I did affected what the other(s) did

Appendix C

The System Usability Scale

One how the most famous questionnaire regarding usability used to have a broad evaluation on the usability

1. I think that I would like to use this system frequently

2. I found the system unnecessarily complex

3. I thought the system was easy to use

4. I think that I would need the support of a technical person to be able to use this system

5. I found the various functions in this system were well integrated

6. I thought there was too much inconsistency in this system

7. I would imagine that most people would learn to use this system very quickly

8. I found the system very cumbersome to use

9. I felt very confident using the system

10. I needed to learn a lot of things before

11. I could get going with this system

Appendix D

VR Usability Evaluation

Module of the VRUSE questionnaires used to evaluate the usability of the application regarding also the context.

D.1 Functionality

- 1. The level of functionality (control) provided by the system was appropriate for the task

- 2. The functionality provided by the VR system was ambiguous(i)

- 3. I found it easy to access all the functionality (control) of the system

- 4. It was difficult to remember all the functions available(i)

- 5. I understood the meaning of the control interface

- 6. I did not need to use all the functions provided(i)

- 7. Overall I would rate the VR system in terms of functionality as: very satisfactory, satisfactory, neutral, unsatisfactory or very unsatisfactory.

D.2 Consistency

- 48. The VR system behaved in a manner that I expected 49. It was difficult to understand the operation of the interface(i)

- 50. The information presented by the system was consistent

- 51. I was confused by the operation of the system(i)

- 52.The sequence of inputs to per- form a specific action matched my understanding of the task

- 53. The use of icons, menus and toolbars was inconsistent(i)

- 54. The actions of controls with icons and symbols were obvious

- 55. Overall I would rate the consistency of the system as: very satisfactory, satisfactory, neutral, unsatisfactory or very unsatisfactory.

D.3   Flexibility

- 56. I found it easy to perform tasks in the order I chose

- 57. The user interface interfered with the way I wanted to interact with the system(i)

- 58. The user can tailor the system to suit their needs

- 59.I could not achieve what I wanted in the VR system (i)

- 60. I was able to take short cuts in using the system

- 61. Overall I would rate the flexibility provided by the system as:very satisfactory, satisfactory, neutral, unsatisfactory or very unsatisfactory.