

Title	Virtual embodied telexistence : telecommunication using sensory feedback and virtual body representation
Sub Title	
Author	Saraiji, Mhd Yamen(Minamizawa, Kota) 南澤, 孝太
Publisher	慶應義塾大学大学院メディアデザイン研究科
Publication year	2014
Jtitle	
Abstract	
Notes	
Genre	Thesis or Dissertation
URL	<a href="http://koara.lib.keio.ac.jp/xoonips/modules/xoonips/detail.php?koara_id=KO40001001-00002014-0392">http://koara.lib.keio.ac.jp/xoonips/modules/xoonips/detail.php?koara_id=KO40001001-00002014-0392</a>

Master's thesis  
Academic Year 2015

Virtual Embodied Telexistence: Telecommunication  
using Sensory Feedback and Virtual Body  
Representation

Graduate School of Media Design,  
Keio University

MHD Yamen Saraiji

A Master's Thesis  
submitted to Graduate School of Media Design, Keio University  
in partial fulfillment of the requirements for the degree of  
MASTER of Media Design

MHD Yamen Saraji

Thesis Committee:

Associate Professor Kouta Minamizawa	(Supervisor)
Professor Akira Kato	(Co-supervisor)

Abstract of Master's Thesis of Academic Year 2015

## Virtual Embodied Telexistence: Telecommunication using Sensory Feedback and Virtual Body Representation

Category: Science / Engineering

### Summary

With the advancements in Internet based services, the demands for better communication tools became higher in order to fill the distance gaps between people. Especially in our daily activities when we are traveling abroad. The traditional video/audio communication tools became actively available within our reach thanks to smartphone advancements. Internet based conferencing such as "Google Hangouts" made it more reachable by using web-based tools, allowing two or more participants to engage in a social meeting. However, the demand for social tools which allows the participants to move freely in the remote place existed. On one hand, Telepresence robots stepped into the scene to fill that need, also it opened new services to the consumers such as remotely visiting museums or attending exhibitions and conferences. On the other hand, Telexistence systems provided more sophisticated capabilities to the user by replicating several functions of his body into the remote place using mechanical parts. "Virtual Embodied Telexistence" is a system that combines low cost telexistence system with virtually embodied human functions into it. Allowing the user to have an immersive visual feedback of the remote place while being aware of his body via visual representation of it.

"Virtual Embodiment" proposed in this thesis defines the important elements for replacing physical representations of human's body into a virtual representation, as a way to increase the sense of presence in a different place. Increasing the sense of body presence in telecommunication helps the user to act more naturally and intuitively using his body motion.

This thesis explores the proposed concept of Virtual Embodiment for human's body and its tight link with presence. The design flow to realize a virtually embodied telexistence system is discussed, as well as the implementation procedure that has been done. We show the efficiency of using this system by objective user evaluation, and how intuitive it is to be used by any user.

**Keywords:**

Telexistence, Computer-Mediated Communication, Reality-Based Interaction

Graduate School of Media Design, Keio University

MHD Yamen Saraiji

# Table of Contents

<b>Abstract</b> . . . . .	ii
<b>Table of Contents</b> . . . . .	iv
<b>List of Tables</b> . . . . .	vii
<b>List of Figures</b> . . . . .	viii
<b>Acknowledgements</b> . . . . .	xi
<b>1 Introduction</b> . . . . .	1
1.1 Computer Mediated Communication . . . . .	1
1.1.1 Telepresence and Telexistence . . . . .	2
1.1.2 Ubiquitous Telexistence . . . . .	3
1.2 Reality-Based Interaction . . . . .	4
1.3 Embodiement in Telecommunication . . . . .	4
1.3.1 Our Body . . . . .	5
1.3.2 Proprioception and Body Schema . . . . .	6
1.3.3 Virtual Reality and Body Representation . . . . .	7
1.3.4 Body in Teleoperation . . . . .	7
1.4 Toward a Social Oriented Telexistence System . . . . .	9
1.5 Thesis Outline . . . . .	10
<b>2 Design of Virtual Embodied Telexistence System:</b> . . . . .	11
2.1 Perception of Presence . . . . .	11
2.2 Measuring the Personal Presence . . . . .	13
2.2.1 Immersion . . . . .	13

*Table of Contents*

---

2.2.2	Involvement . . . . .	14
2.3	Design Considerations . . . . .	15
2.3.1	Remote Accessibility . . . . .	15
2.3.2	Control and Navigation . . . . .	15
2.3.3	Embodiment . . . . .	17
2.3.4	Virtual Embodiment Elements . . . . .	18
2.4	Designing Virtually Embodied hands . . . . .	19
2.4.1	Simulator Design . . . . .	19
2.4.2	Hands Superimposing . . . . .	20
2.4.3	Preliminary Results . . . . .	21
<b>3</b>	<b>System Implementation . . . . .</b>	<b>22</b>
3.1	System Overview . . . . .	22
3.2	Telexistence Robot Implementation. . . . .	25
3.2.1	Head Design . . . . .	25
3.2.2	Embedded System . . . . .	27
3.2.3	Vision Correction . . . . .	28
3.2.4	Mobility Design . . . . .	30
3.3	Operator System . . . . .	35
3.3.1	Tracking System . . . . .	36
3.3.2	Virtual Embodied Arms . . . . .	39
3.3.3	Haptic Feedback System . . . . .	41
3.4	Communication System . . . . .	46
3.5	User Interface . . . . .	48
3.6	Technical Evaluation . . . . .	49
<b>4</b>	<b>Evaluation and User Study . . . . .</b>	<b>52</b>
4.1	Evaluation of Presence . . . . .	52
4.1.1	Participants . . . . .	53
4.1.2	Environment Setup . . . . .	53
4.1.3	Experiment Procedure . . . . .	54
4.1.4	Behaviour Analysis . . . . .	61

*Table of Contents*

---

4.2	Public Subjective Evaluation . . . . .	62
<b>5</b>	<b>Reserch Contribution, Limitations, and Conclusion . . . . .</b>	<b>64</b>
5.1	Research Contributions . . . . .	64
5.2	System Limitations . . . . .	65
5.3	Conclusion . . . . .	66
	<b>Bibliography . . . . .</b>	<b>68</b>
 <b>Appendices</b>		
<b>A</b>	<b>Software Implementations . . . . .</b>	<b>73</b>
A.1	Class Diagrams . . . . .	73
A.1.1	User Class Diagrams . . . . .	73
A.1.2	Robot Class Diagrams . . . . .	73
A.2	Code Listings . . . . .	77
A.2.1	Camera Lens Correction . . . . .	77
A.3	GStreamer Pipelines . . . . .	78
A.3.1	Video Pipelines . . . . .	80
A.3.2	Audio Pipeline . . . . .	81



# List of Tables

3.1	Head joints angle limits. . . . .	27
3.2	Specifications of two mobile platforms used in the system . .	30
3.3	Control signals used between user and robot . . . . .	47
3.4	Video quality requirements for the robot and user side . . .	48
3.5	The average latency in millisecc using two video streaming resolutions and several bitrates . . . . .	51
4.1	Set of questions before starting experiment . . . . .	56
4.2	Set of questions after first phase: Controlling the robot . . .	57
4.3	Set of questions for Group A and B after phase 2 - No hands are presented . . . . .	58
4.4	Set of questions for Group A and B after phase 3 - Hands are presented . . . . .	59
4.5	Set of questions after the end of the experiement . . . . .	60

# List of Figures

1.1	(a) Beam and Double Robotics Telepresence Robots, (b) TELESAR 2 Telexistence Robot . . . . .	2
1.2	Body representation in Telexistence and Telepresence. Left: TELESAR V, Right: Rollin' Justin . . . . .	8
1.3	Telepresence robots used for social communication with remote participants. . . . .	9
2.1	The three levels of perception . . . . .	12
2.2	CAVE and HMD VR systems . . . . .	14
2.3	The proposed robot motion control. . . . .	16
2.4	See-through HMD. . . . .	20
2.5	Real hands captured using STHMD, Generated Mask, and superimposed hands in VE. . . . .	21
3.1	Enforced Telexistence System Overview. . . . .	23
3.2	(a) A user looking through robot eyes in a different place, (b) Robot placed in a different environment, (c) First Person View of what the user sees. . . . .	24
3.3	System distribution for ubiquitous access with the data flow to the user. . . . .	25
3.4	(a) Major components of robot's head, (b) CAD model of the head robot, with axis names. . . . .	26
3.5	Intel NUC board with the corresponding connections . . . . .	28
3.6	Placement of the processing system in correspondence to head	28

*List of Figures*

---

3.7	Webcamera used in the head implementation, and the three tested types of lens . . . . .	29
3.8	The two implemented versions using Omni Wheel and Roomba	31
3.9	Sensors used for collision avoidance. . . . .	32
3.10	Major components for display and tracking. . . . .	35
3.11	Illustration of HMD motion based control. . . . .	37
3.12	The captured IR images of the hands with joint information for each finger . . . . .	38
3.13	(a) Virtual 3D hands superimposed on remote environment. (b) Using the hands to touch objects. (c) Superhuman ability, extending the arms. . . . .	40
3.14	(a) Captured hands from Leapmotion (b) Processed image, remove background (c) Final superimposed hands . . . . .	40
3.15	Processing the captured depth map to generate surface map	43
3.16	Estimating touch and stroke forces on user fingers using SM	45
3.17	Touch feedback display . . . . .	45
3.18	User interface depending on the scenario. . . . .	49
3.19	Leapmotion framerate comparision while running on different modes. . . . .	50
3.20	Video stream latency over time . . . . .	51
4.1	Number of participants used HMD before. . . . .	53
4.2	Experiemnt environment setup (a) Robot room. (b)User room. . . . .	54
4.3	Phase 1 Experiemnt, User controlling the robot using his body motion. . . . .	56
4.4	Phase 3 User experience his hands, and uses them to communicate. . . . .	58
4.5	Results.of the evaluation of GroupA and GroupB. . . . .	60
4.6	Changes of participants state before and after the experiemnt	61

*List of Figures*

---

4.7	Users experiencing Out Of Body and trying to touch their own body from out side (SIGGRAPH Asia 2014). . . . .	62
4.8	People with different ages testing the system (KMD Forum 2014). . . . .	63
5.1	TORSO 6DOF Telexistence system. User experience virtual embodiment using TORSO system. . . . .	65
A.1	User side main components class diagram . . . . .	74
A.2	Robot communication handler class diagram . . . . .	75
A.3	Robot control class diagram . . . . .	76
A.4	Video/Audio Streamers Class Diagram . . . . .	78
A.5	Video/Audio Players Class Diagram . . . . .	79

# Acknowledgements

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

First and foremost, I thank God for being my strength and guide in doing this research and writing the thesis.

I express my deep gratitude to my supervisor Prof. Kouta Minamizawa for all his assistance and critical academic feedback along the year. I am thankful for his patience, motivation, and immense knowledge. I am greatly thankful for the great guidance that I received from him.

I was very honored and lucky to have direct feedback and comments about my research from Professor Emeritus Susumu Tachi. His immense knowledge of over 30 years in Telexistence and Haptics was a great inspiration for me. Also I would like to thank the dean of KMD Prof. Masa Inakage for his great thoughts and inspirations along the year, and for his support.

Also I would like to thank my co-supervisor Prof. Kato Akira. I got a lot of academical and technical knowledge support from him. And for Prof. Masahiko Inami, I am deeply thankful for providing me with many feedbacks to improve my research, and providing me several knowledge related to this work.

I would like also to have the chance here to thank my previous advisor, Masahiro Furukawa. I thank him for his very serious attitude in dealing with the research work and for his high quality standards in presenting the research. He was a great teacher, and great support when needed. Both in research and personal life.

## *Acknowledgements*

---

I would like to give a special gratitude for my senior Charith Fernando for devoting much of his time in providing me direct support during the research and the academic journey in KMD, and for his great companionship along the year. Without him, this research wouldn't have the light to finish with the presented quality.

During the period of the research and the implementation, I received many precious support especially in some technical related matters. In this context, I would like to thank Youichi Kamiyama san for his high quality work in manufacturing some hardware parts for the robot. Also I would like to thank Yusuke Mizushina san for his help in completing the haptic display, and his support in doing the public demonstration at SIGGRAPH Asia 2014. With his help, I managed to complete the system and presenting it to many professional attendees.

I want to thank the past and present members of Tachilab group and Livinglab for their support and sharing with me their knowledge in the field of Telexistence, Haptics, and Interaction Design.

Tons of thanks goes to my dear family who were very supportive in all aspects of personal life, and their patience for my long absence, especially during the current catastrophic back home in Syria. I thank my mother for the ethics she taught me, and for her supportive attitude to persuade my dreams and goals. And to my father who raised me to be a strong, independent man. They supported me and encouraged me along this research. I hope I have made them proud. And I miss them too much.

*The Author*

*MHD Yamen Saraiji*

# Chapter 1

## Introduction

*“Each of us believes himself to live directly within the world that surrounds him, to sense its objects and events precisely, and to live in real and current time. I assert that these are perceptual illusions ... Each of us lives within the universe - the prison of his own brain”*

*Vernon Mountcastle- The View from Within*

### 1.1 Computer Mediated Communication

Telecommunication over the Internet has experienced many changes and improvements since the very first basic usages of it, the text-based Computer Mediated Communication (CMC) [1]. Though until now we still rely on such basic methods with a combination of recorded media, for example in our email or in social networks. As technologies change, so the CMC forms evolve. This evolution was noticeable by how we started with text-based communication to send text and basic media (such images) messages when the network capabilities were limited in speed and bandwidth, then it was shifted into voice-based telecommunication which delivers much complex verbal messages, and now using video/audio communication with high definition quality in order to provide a better sense of communication with others. Generally CMC media can be Synchronous and Asynchronous, however here I will focus on the Synchronous type, since the other does not fall in the scope of this thesis.

The richness of CMC environment that contributes in delivering non-

verbal information plays an important role at providing sense of presence in a social activity [2]. As the social interaction between people is important in CMC [3], highly interactive contents are necessary to be defined and created that participants can access within CMC's cyberspace. Using these contents, the participants can have higher sense of presence between each other.

### 1.1.1 Telepresence and Telexistence

To expand the sense of presence of a participant in CMC, the utilization of Virtual Reality (VR) was proposed and used in telecommunication. In 1980, two concepts have emerged in the West and Japan: Telepresence [4] and Telexistence [5]. Both share the idea of delivering real-time physical presence of one being into other place by using a slave robot controlled by the user, an example of each system is shown in figure 1.1. Basically, Telepresence focuses on the means that the user adopt to real-time control a remote robot in a different place, the robot in this case does not have to share physical attributes with the user, such as stereovision, or humanoid structure. Telexistence in contrast, focuses on replicating some of human

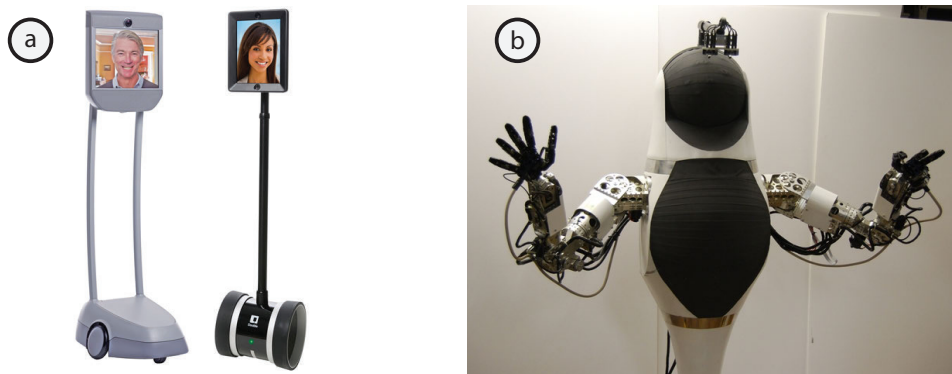


Figure 1.1: (a) Beam and Double Robotics Telepresence Robots, (b) TELE-SAR 2 Telexistence Robot

Photo © Suitable Technologies, Double Robotics, and Tachilab

functions in other place, and deliver real-time sensation of being at a dif-



ferent place. In this case, the visual, auditory and haptic perceptions are necessary to deliver to the user from the remote robot. Telexistence robots are usually accessed via a Head Mounted Display (HMD) that provides visual feedback from the robot, and controlled via a set of tracking tools mounted on user's body.

### 1.1.2 **Ubiquitous Telexistence**

The idea of ubiquitous telexistence systems is to distribute telexistence systems among wide area, such as world wide. Allowing people to “teleport” to those distributed points for the sake of visiting, or operating a specific task. Realizing this concept requires defining the major physical components that each system should provide. The requirements may change based on the target application that this network of systems should provide. Perhaps for a sightseeing purpose, a minimum system that has the ability to move and to provide first point of view (FPV) visuals are required, with no need for any articulated arms or body. However, for functional purposes that requires manipulation, then those arms and other limbs are required.

Usually when social interaction is required (talking with other people, communication in general) the body language and mutual communication is required in this case. Also the sense of bodily presence is needed for the user when a communication is presented. Providing such functions is reflected on the design of the target ubiquitous robot. In this case, it is a challenge to design a telexistence robot with such functions, since there is a trade off between the functions and the cost/complexity to build such a robot.

## 1.2 Reality-Based Interaction

Human Computer Interaction has observed dramatic changes and evolution from the Command Line to Direct Manipulation via "Window, Icon, Menu, Pointing Device" (WIMP) style, and now via body based interaction using gesture tracking devices, and body tracking. These generations of interaction started to move computer interfaces more closer into the real-world interactions, allowing the user to naturally access and manipulate digital information. A framework was proposed in [6] that defines the boundaries of RBI.

Creating a seamless interactions which maps our native physical knowledge into the virtual world is one of the main topics in HCI. The interactions from the real world around us should be the starting point to define and create future's computer interaction [7]. One of the very basic knowledge we know how to use is our body in general, and our hands in specific. Adopting our body into virtual worlds and cyberspace would provide wider accessibility and more easiness as well as a higher sense of actually "being" in that cyberspace.

RBI is this wide framework that contains the concepts and tools we have in our physical world, and combine them into virtual medium. This medium creates a seamless connection with our real-world, and the way we interact with this medium and its function does not conflict much with the knowledge we know from our physical world.

## 1.3 Embodiment in Telecommunication

Telecommunication media development has redefined our social and day to day activities. Media interfaces are progressively growing to embody us within it, affecting our sensations of physical presence, social presence, and self presence inside. This progression in telecommunication technologies

creates a tighter coupling between the body and the interface, in which the body becomes a part of the physical space and the cyberspace. This progression creates an adaptation between the body and the interface [8].

### 1.3.1 Our Body

Our bodies are the connection to the reality and the physical space surrounding us, with which we can participate in every day activities. The presence of a body is so obvious that its main functions can be listed as [9]:

- The physical embodiment of oneself.
- Core sensors of our mind: Perceiving our world through it, and mediating it to our brain.
- A medium of interaction: Capability to change our world.
- A medium of communication: Using the gestures and sound to interact with other people.
- Social representation of self: Our bodies are the mirror of individuals, through them we recognize their presence and their behavior.

The link between our bodies and sensory perception is what makes us understand the reality. Though in the ancient times, it was held the belief of our bodies prevent us from knowing the reality:

“And I suppose it <sup>1</sup> reasons best when none of these senses disturbs it, hearing or sight, or pain, or pleasure indeed, but when it is completely by itself and says good-bye to the body, and so far as possible has no dealings with it, when it reaches out and grasps that which really is.” - Socrates <sup>2</sup>

---

<sup>1</sup>“it” refers to the soul

<sup>2</sup>Plato, Phaedo.

Nowadays, it became trivial and fundamental the body presence and sensory inputs are the key to understand our reality and create a knowledge about the world surrounding us [10, 11].

Understanding our bodies in a social context can be defined as an expressive communication device [12] that represents internal mental states (e.g. thoughts, emotions, etc.) to others.

### 1.3.2 Proprioception and Body Schema

Proprioception has been defined by neurologist Sacks [13] as :

“... that continuous but unconscious sensory flow from the movable parts of our body (muscles, tendons, joints), by which their position and tone and motion is continually monitored and adjusted, but in a way which is hidden from us because it is automatic and unconscious”

Proprioception creates a mental model of our body, which describes the body parts in relation to each other. So we can know the location of our feet without looking at it, or touching or ear with closed eyes. This kind of mental model was developed since we became sensory aware of our body.

Body Schema involves aspects from both brain processing and body peripheral (Proprioception signals, sensory signals), and it considered as a collection of processes that stores the posture of the body. When the body state became in motion, the body schema updates unconsciously according to that motion [14].

People with phantom limb pain (after losing one part of their body, like an arm), their brain continues to detect the presence of their missing limbs even though its no more attached. This type of phenomena is more likely related to proprioception signals in the brain [15]. Since their body schema has changed, they need time to adapt to the new schema.

### 1.3.3 Virtual Reality and Body Representation

Virtual Reality has presented a new form of presence in a different place. The sensory integration and feedback (head tracking, motion parallax, realistic visuals) can trick the mind and makes it believe of being inside that world. However when one tries to look at his own body and finds either no representation of it or a computer graphics representation, the illusion of presence is immediately dismissed. The need of presenting a virtual body representing our own is to match the proprioception signals with the sensory feedback and reduce the contradiction between both.

### 1.3.4 Body in Teleoperation

Teleoperation systems does not differ much from Virtual Reality in terms create the sense of presence in a different place. The same considerations faced in VR applications are also applied in Teleoperation. Strictly speaking, body schema mapping between the user and the robot should match to a certain degree to avoid the lack of sense of presence in the remote place.

For applications that require direct manipulation with physical objects, a physical representation of human limbs are usually implemented with different degrees of freedom for operation. Usually in Telepresence application, a very basic reproduction of human body is done, and body schema can be different from what human body schema is actually is. So the number of degrees of freedom (DoF) can be less than DoF of human's body, like less number of fingers each hand has and replacing the full hand with a grip like end effector. In Teleoperation applications that uses telepresence for its purpose, the operator usually train for a long period in order to adapt to the new body schema of the robot. Controlling of this new body schema also will require the user to learn it.

In Telexistence based applications, usually an exact (almost exact) physical body schema of human body is used. Replicating human's sensors like

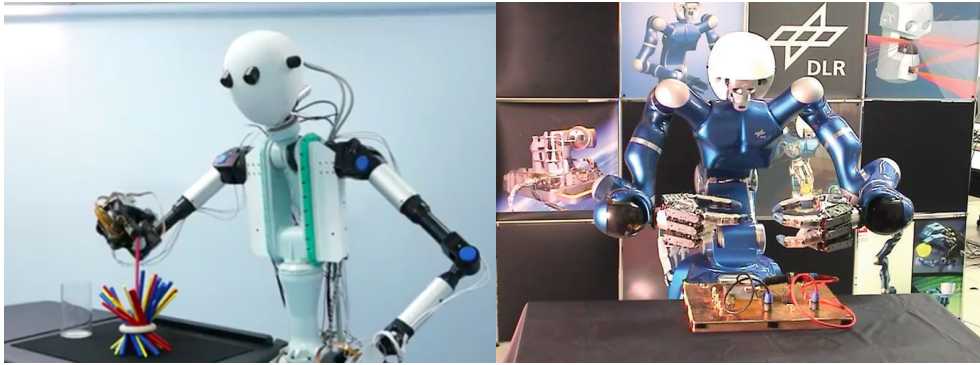


Figure 1.2: Body representation in Telexistence and Telepresence. Left: TELESAR V, Right: Rollin' Justin

Photo © Tachilab and DLR

binocular vision, stereo audio feedback, head degrees of freedom, and haptic feedback. Figure. 1.2 shows two humanoid systems used for Telexistence and Telepresence applications, it can be noticed the level of resemblance to human body in Telexistence robots.

On one hand, Telexistence type of robots has been mainly used in applications that requires human's expertise in the work place. Offering the user a semi-replication of his body using sophisticated mechanical representation of his arms and hands. On the other hand, Telepresence robots has been dominantly focused on social experience of the user in other place Figure. 1.3, providing him the means to "See", "Talk/Hear", and "Move" remotely. However Telepresence compared with Telexistence does not provide the user sense of body presence in the remote environment .

Adapting Telexistence systems into social environments creates a challenge both in design and engineering. In other words, to be able to create a system thats small and cost-effective, and provide the user enough embodiment so he can experience of being in the other place.



Figure 1.3: Telepresence robots used for social communication with remote participants.

Photo © Suitable Technologies and iRobot Ava 500

## 1.4 Toward a Social Oriented Telexistence System

To realize a telecommunication platform that enable the person to have the feeling of complete presence in another place, all the visual, auditory, haptic feedback from the remote place should be provided, as well as user's body visuals and interactions in the remove place. The combination of the physical body representation and the remote feedback increases the sense of physical presence, and reduces the sense of distance between the local and remote. Enabling visuals feedback of the user's body, the user would have higher sense of presence in the remote place and behave more naturally.

This research studies the elements of presence, and the effect of body existence on the quality of communication. We propose here a hybrid approach to replicate human's body in a different place. The approach combines physical and virtual representations for human's body , allowing

the user to expand his sense of presence into a remote place. We introduce two types of visual presentation of the body:

1. Computer generated human body visuals.
2. Image-based human body visuals.

Both of the two previous methods uses different technique to provide body visuals feedback while operating in remote place. Also we present here a novel technique for presenting fingers touch haptic feedback with remote objects. This method does not require any physical touch sensors in the remote robot, and only uses depth array sensor instead.

## 1.5 Thesis Outline

This thesis is structured to several chapters, and each is described as follow:

- Chapter 1 has presented some background researches, the driving problem, and motivation regarding this research. As well as related approaches to provide RBI in CMC systems.
- Chapter 2 discusses the design and thinking methodology of this system. Highlighting the important points of the target experience for the users, as well as the design of the interaction. Early prototypes and concept proofing results will be listed too. Also in this chapter, some systematic design consideration will also be highlighted.
- Chapter 3 will explain in details the implementation process of this telepresence system based on the design considerations. The hardware and software details will be shown in this chapter.
- Chapter 4 describes the user study of the proposed system, the results of the evaluation will be discussed.
- Chapter 5 lists the main contribution points, and concludes this research.



# Chapter 2

## Design of Virtual Embodied Telexistence System:

### 2.1 Perception of Presence

The perception of presence for a human being defines how he perceives the environment surrounding him via human's sensory in order to provide the belief of being at a certain place [16, 17]. Actually presence has two definitions commonly used by researchers in this field. One is defined as the subjective experience of being in one place or environment, even though physically being situated in another place [18]. As for example, in teleoperation tasks it is described as the sensation of being in the remote working site rather than the physical operation control site. The other definition is the perceptual illusion of non-mediation [19]. In other words, when the user involved in an experience fails to acknowledge that it is being presented to him through some sort of media due to high level of sensory feedback (e.g. IMAX, HMD).

In this system, the proposed Telexistence system aims to deliver to the users a high level of presence in a different place, and in order to do that, the sense of presence is divided into three levels: personal presence, environment presence, and social presence. Each of those involves to a different aspect of the experience in order to improve it.

For the first level "Personal Presence", we understand the presence of our bodies via the multisensory integration. In the general case, our body

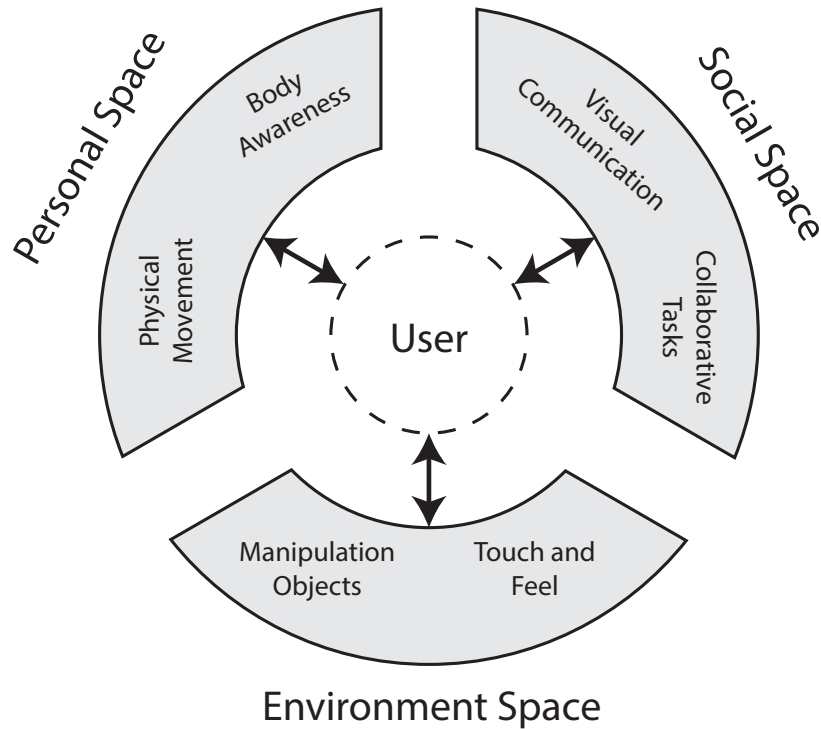


Figure 2.1: The three levels of perception

belongs to our local spatial domain (the physical body parts), but also the same experience can be applied to an alien body limbs via cross modality perception. Rubber hands illusion is an example of using visual and haptic feedback cross modality integration [20].

The second level of perception is “Environment Presence”, in which it defines how the user perceives the environment surrounding him. In this level, the *involvement* and *immersiveness* with the surrounding objects determines whether we are presented in a specific place or not. The multi-sensory integration and variety (e.g. touching what we see) are also important factors in defining the level of immersiveness. Usually Virtual Reality (VR) applications maximize the number of elements involved in the experience (e.g. large field of view, motion tracking, haptic feedback, etc ...) to create the belief of presence in the artificial environment.

The third level of perception is “Social Presence”. This level is only relevant in cases which humans interactions are involved between each others. It defines what people uses or do to engage in a social communication as a group. By using verbal and non-verbal communication, people collect information or form feelings toward each others [21].

In this research, the proposed CMC Telexistence research aims to create a social involving telexistence robots that exposes to the user the three levels of perception to create a mobile, low cost, telecommunication platform with a believable sense of presence in a different place. Those three levels are shown in Figure 2.1.

## 2.2 Measuring the Personal Presence

In order to measure the level of personal presence for a user in a VE experience or using a telecommunication system, two factors are used: *Immersion* and *involvement*.

### 2.2.1 Immersion

The immersion in an experience is a psychological state that is described as perceiving oneself being included in ,surrounded by, and interacting with an environment that provides stimuli and experiences toward the person. An environment that provides higher sense of immersion, will produce a higher sense of presence. The immersion is the affect of isolation from the physical place in which a user is physically situated at, and the perception of self-inclusion in a different environment. Factors that affect the immersion include the perception of self-movement (head movement, body movement), natural interaction with the surrounding environment, and the sensory feedback. Usually head-mounted displays (HMD) are used as a device to provide isolation in the VEs. When users perceive them-

selves as being physically outside the environment (for example, by looking through a desktop display), then the immersion is lost with that environment. HMDs are not the only devices used to provide immersive VE, CAVE systems [22, 23] are also used to provide an immersive experience by projecting VE images on walls around the user, usually stereo images are project with equipped shutter glasses. Figure 2.2 shows the two VR systems in applications related to visualization and in games.

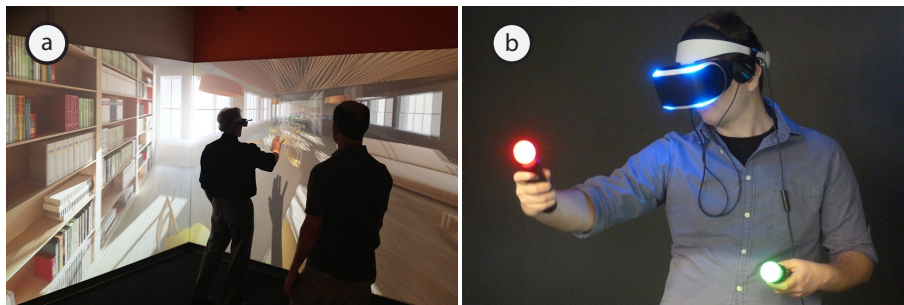


Figure 2.2: CAVE and HMD VR systems  
Photo © WorldViz and Sony

### 2.2.2 Involvement

Involvement is described as the psychological state of a person after being engaged and focused in a stimuli or meaningfully related activities and events. In general, when people focus on the VE stimuli experience, they become more involved in this experience, which increases the sense of presence in that environment. Distraction is the main factor of decreasing the level of involvement for the user, such as being in an uncomfortable position, or uncomfortable HMD (especially in VE applications and Telexistence). Also, the involvement is decreased in case of mismatching the mental model of the user (his expectation), and what he actually observe or sense (the end result).

In teleoperation and Telexistence applications, the involvement factor is a necessary element for the user to gain the believe of being engaged in the

operation or the communication with other participants in a remote place.

## 2.3 Design Considerations

### 2.3.1 Remote Accessibility

The design of this system takes into the considerations the capability to connect and have a remote access into a different place via the proposed robot, fulfilling the ubiquitous design requirement. With a remote control and sensory feedback, latency related issues become important to address. The user should have a minimum notable delay during interaction, especially regarding user's body movement and the visual input that he observe.

### 2.3.2 Control and Navigation

To provide the ability to move freely in a different place is a consideration for both the local side (user) and the remote side . Remote side includes both the robot and other participants around the it.

Considering the local side, user's interactions are highly important to be considered for controlling the robot movement. In teleoperation applications, the user is usually exposed into 2D screen surface that provide robot's vision, and a set of tools to control the movement and operation of the robot. Joystick based movement control has been commonly used in several teleoperation applications, however they lack the intuitive controlling and they usually need training ahead. Other direction for controlling the motion in VR application is gesture-based interaction [24, 25]. In this approach, the user uses his hands to navigate and do actions in the VR space. A common drawback of this type of interaction is overloading the hands with gestures that need to be learned ahead, and the complexity of the gesture increases by the number of interactions.

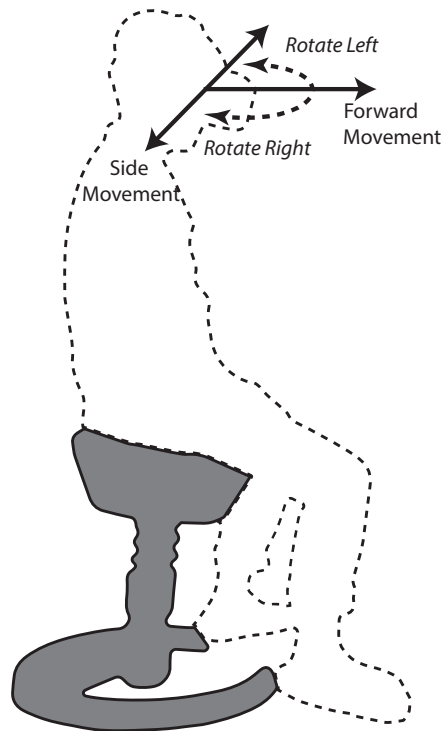


Figure 2.3: The proposed robot motion control.

Humans primarily rely on vestibular and visual cues for balance and orientation. Also we use these senses to determine whether we have self-motion or if the environment around us is moving. In this telexistence system, its necessary that the user perceive the motion as self-motion when navigating. Especially in cases when the robot do turns to left and right. Joystick-based control method lacks cross modality between the visual and vestibular senses. Locomotion-based navigation methods using treadmills [26, 27] provides much natural and intuitive control compared with joystick-based, however using these systems for long time can cause fatigue for the users. A trade of between the two methods is by using body-controller interface [28, 29], in which the user navigate in VE by leaning his body.

In this research, body-controller interface type of navigation was favored over joystick-controller type. The proposed navigation uses head movement to decide moving direction of the robot. Figure 2.3 illustrates the

motion control method. In this type of control, only head position and orientation are taken to consideration to calculate motion vector and rotation speed of the robot. The implementation chapter will discuss in details the procedure.

## Safety Considerations

Though the user will have the means of presence in another place via an avatar robot, the awareness of robot physical body is still limited due to lack of sensory feedback to the operating user (e.g. full body haptics, peripheral vision). With mobility feature in the robot, the robot will have a high probability to collide with environment objects that user can't see or sense (due to lack of peripheral vision), which as a result can cause a damage to the robot, or objects, or even worse for people or children.

Thus in the design of the robot, takes to consideration the safe operation of the robot. To do this, adding sensors (like IR or bump sensor) that scans and determines if there is an object close to the robot. The safety operations will run on the robot level, which means it will have a higher priority than user's commands, and will override them if necessary. More details will follow in the Implementation chapter.

### 2.3.3 Embodiment

Expanding user's sense of presence in the remote environment is achieved by creating the believe of "being there" using his body. When the user see his body being seamlessly merged in the different place using visual, physical and functional manner, he would start believing of actually being in that place. Previous works in Telexistence and Virtual Environments has explored this topic by creating physical or virtual representation of user's body. In virtual environments, Biocca [30] explored the effect of embodiment in virtual worlds on the mind. TELESAR V [31] is an example of

Telexistence systems that uses a humanoid robotic system to realize user's body in a different place, the user observe his body being turned into a mechanical form, allowing him to operate naturally from the first point of view.

In this research, we consider the importance of adding body representation for the user as a factor to create more natural feeling of being physically communicating, and to avoid any feeling of disturbance when the user feels being disconnected from his body. We integrate user's body as a virtual representation, so he can have visual feedback of his body when he tries looking at it. The behavior of the body when trying to touch also important so the user can interact, touch and feel with remote objects.

#### 2.3.4 Virtual Embodiment Elements

In virtual reality applications, the idea of embodiment is to replicate an idea, physical object, or behavior into a digital form that can be access via an interactive environment. Adapting this concept into human's body means to create a matching virtual form of the body. The main elements in designing this digital form can be categorized to the following:

1. **Visual Appearance:** Matching the appearance and the look of the body, so the user as he looks to his body, he would have the believe its his own.
2. **Physical Behaviour:** Natural motion and mapping with the body, as well as preserving body schema. When the user tries to move, his body should follow accordingly and the user would perceive it as expected. Thus no conflicts with proprioception signals would occur, and no motion sickness.
3. **Functionality:** The ability to respond as it is expected to, such as touching and feeling.



The combination of the previous three elements in any VR application let the user have the believe of being physically existed in that environment, and no more in his original place.

## 2.4 Designing Virtually Embodied hands

As for initial design steps of the systems, several functions related to the system design and virtual embodiment methods were conducted in a virtual environment via a custom made simulator. This simulator aimed to minimize the time and cost of testing and iterating before a real system is fabricated. Since the purpose of the simulator is concept testing and proofing in an early stage, it will be discussed in this section no in the implementation. Previous publication by the author [32] regarding to the design of the simulator.

### 2.4.1 Simulator Design

The simulator was meant to isolate the hardware related tasks from software related tasks, and focus mainly on the functions the system provides. In other words, it is designed to test virtual embodiment concept by providing the user the visuals of his body and haptic feedback in a virtual environment.

The simulator basically create a VE in which the user can access via a replicated virtual version of a Telexistence system (Telesar V [31]). Controlling method of this virtual version is exactly the same as controlling the original system. The user in this system wears a set of trackables to measure his body, head and hand movement. For fingers tracking, 5DT Data Gloves were used that uses fiber optics to measure the amount of bending for each finger.

In this early design, a see-through HMD (STHMD) [33] shown in Figure

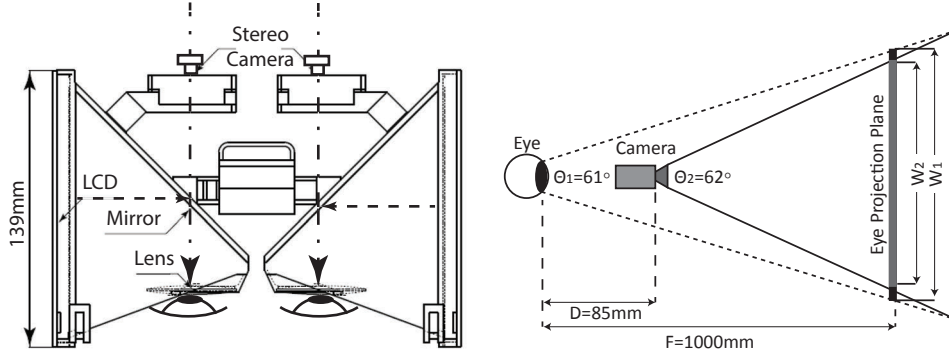


Figure 2.4: See-through HMD.

2.4 was used to capture user’s hands visuals. As can be seen in the figure, some displacement is presented in the design, that results scaling of the hands compared to looking at them with naked eyes. More details about correcting this problem is presented in a previous work by the author [34]. The simulator runs on a single PC, and uses NVidia PhysX for physics simulation. The rendering is stereoscopic and it runs at 60FPS that ensures low latency. It is connected over network to communicate with tracking and haptic control systems.

### 2.4.2 Hands Superimposing

At the initial stage of this research, virtual embodiment was conducted in a virtual environment, the hands also were modeled as virtual hands but the texture of them were driven from operator’s hands.

Hands visuals are captured using STHMD that user have, stereo images of the hands are captured and processed inside the simulator (a). Using calculated user’s kinematics and joint information, a 3D model is constructed and placed in user’s virtual position. Then a stereo mask images are generated (b) and used to cut off the background from camera’s images. Finally, the masked images are used inside the VE (c).

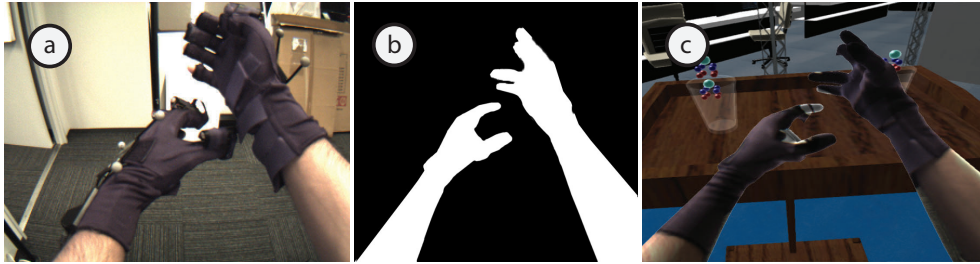


Figure 2.5: Real hands captured using STHMD, Generated Mask, and superimposed hands in VE.

### 2.4.3 Preliminary Results

The initial prototype that is using STHMD and model based experienced several problems as follow:

- Tracking related issues: The body kinematics calculation used faced problems in providing an accurate posture for the user, especially for hands and finger position. These issues are regarded to the limited tracking space, and limited number of finger joints from the data gloves.
- Not accurate masking: As can be seen in the previous figure, some background texture appeared inside the hands. The problem is the difficulty to match model mask with user's body accurately.
- Unrealistic visuals: Due to amount of wearable tracking devices they user have to put, the actual visuals of user's skin were hidden, thus it was not quite fit for the first factor of virtual embodiment: Visual Appearance.

However in general, the outputs of this initial design was helpful to develop a non-wearable, less trackables system. Also it provided an initial experience of the virtual embodiment in a different space.

# Chapter 3

## System Implementation

In the previous design chapter, several important points were highlighted regarding the requirements and specification to implement a virtually embodied telepresence system. This chapter will discuss in details the implementation process of the proposed system, as well as the tools used during the process.

### 3.1 System Overview

Encorced Telepresence System is divided into two main systems: Telepresence Robot System that resides in the remote place, and User's Control System in the local operating place. A general overview of this system can be shown in Figure 3.1. In the user side (local), a set of tracking tools are used to capture user's head movement (Oculus DK2) as well as hand movement (Leapmotion). Head rotational motion is then mapped into the remote robot's mechanical head. Also head linear movement (translation) is used to control robot's speed, further details regarding robot movement control will be discussed in this chapter. The captured Hands movement and visuals are used to provide virtual embodiment to user's side. The user observe his own hands motion over robot's vision. Also haptics feedback is provided on user's finger when touching remote objects.

In the robot side (remote), a 3 Degrees of Freedom (DOF) head is used to physically map user's head rotational motion in the remote place. FullHD stereocameras are used as well as binaural microphones. The robot provides

### 3.1. System Overview

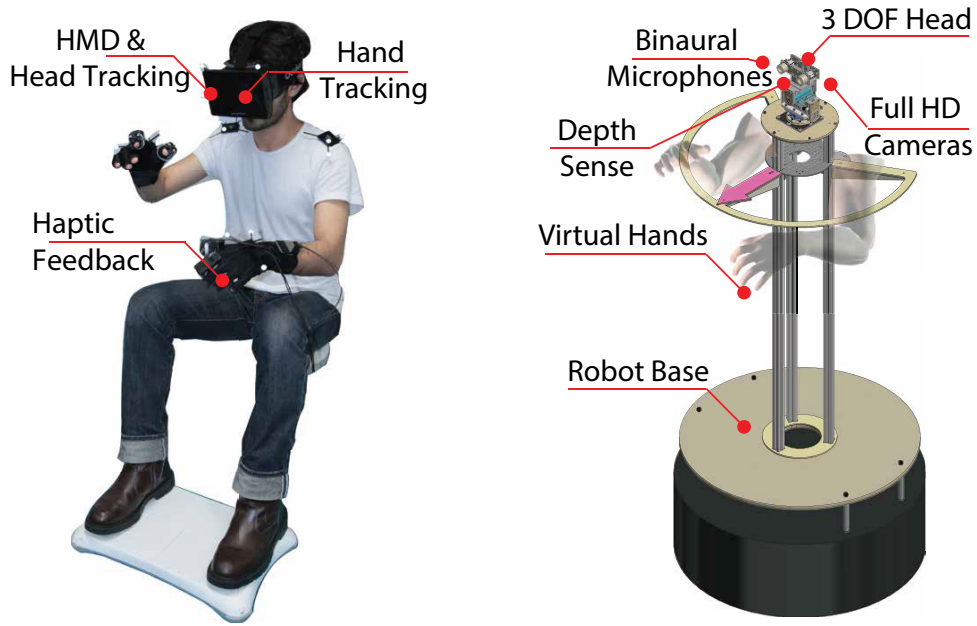


Figure 3.1: Enforced Telexistence System Overview.

user's voice in the remote place via a speaker mounted on the robot. Also it has a LCD mounted on it that shows the operating user's top body and arms. The robot designed over a mobile platform (Two implementations: Omni robot, iRobot) that allows it to move freely in the remote place. The robot height is manually adjustable, and it can range from 120cm to 170cm, typically the height is fixed at 130cm for best interactions with people standing and setting.

The robot system runs over WiFi connection to provide complete mobility. In Figure 3.2 (b) shows the system being operated remotely by the user in (a). The user can observe the remote objects and environment, as well as preceiving his body visuals and motion being virtually embodied in that remote place as shown in (c).

The design of the system also aimed for ubiquitous access to the telexistence robots, Figure 3.3 shows an example of distributed virtual embodied telexistence platforms. The software design of the system realizes this goal,

### 3.1. System Overview

---



Figure 3.2: (a) A user looking through robot eyes in a different place, (b) Robot placed in a different environment, (c) First Person View of what the user sees.

### 3.2. Telexistence Robot Implementation.

so all the connections between the user and the robot are running in a networked environment, no direct wired connection between them exist.

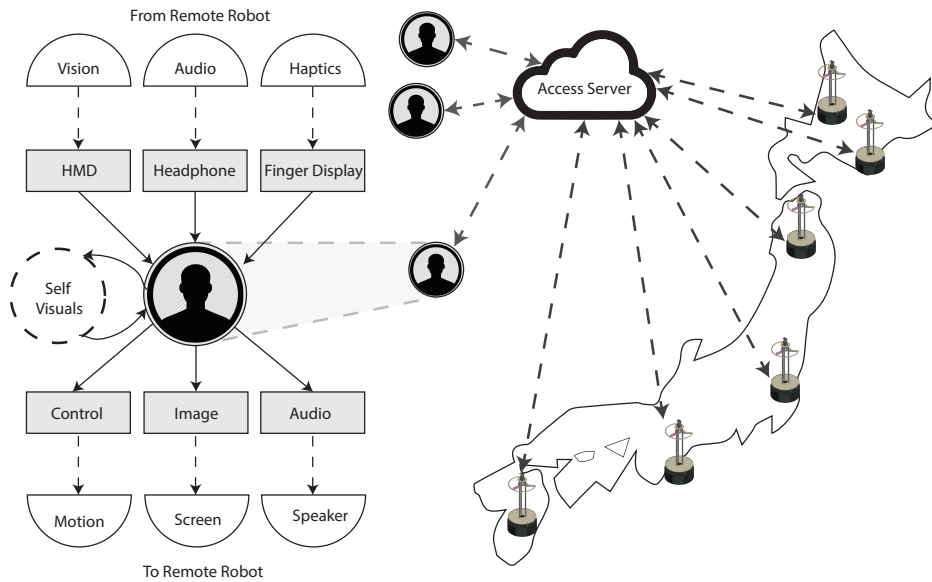


Figure 3.3: System distribution for ubiquitous access with the data flow to the user.

## 3.2 Telexistence Robot Implementation.

In this section, a detailed description about the implementation of robot side is shown. The main components which defines this Telexistence robot are the head and mobile base. Also, the robot provides a basic mutual interface that gives feedback of the operator using the robot.

### 3.2.1 Head Design

The design of robot's head is aimed to match the major properties and functions which a human head has. The main functions are listed as follows:

1. Head Motion: 3 head axis (Pitch, Yaw, Roll).

### 3.2. Telexistence Robot Implementation.

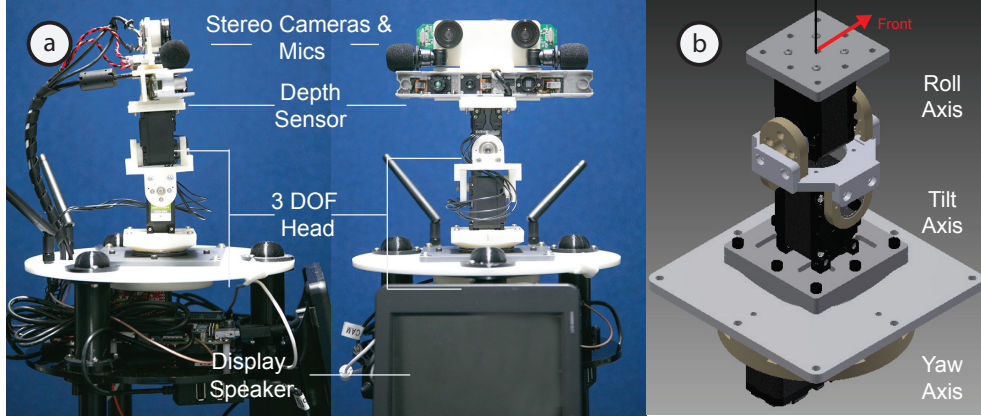


Figure 3.4: (a) Major components of robot's head, (b) CAD model of the head robot, with axis names.

2. Stereo Vision: Two eyes with a fixed interpupillary distance (IPD).
3. Binaural Sound: Two ears.
4. Audio: Human mouth.

The previous functions are the fundamental elements to replicate user's head. One more component was also added to scan the remote place geometry. A depth array sensor was placed at the eyes level. The function of this sensor is related with estimating haptic feedback on the user's fingers (further details will be discussed in this chapter). The final head design with the installed components, as well as the 3D design of the head model can be seen in Figure 3.4 <sup>1</sup>.

The head motion is a high-speed with high-torque servo motors (Model no: HerkuleX DRS-0201) were used for the three motion axis of the head. The motors are controlled via a motor drive controller (Model no: AT-MEGA8U2 Breakout Board). The three servos represent head rotation on the three axis: Pan, Tilt, and Roll. Each of those joints has an angular limit, the values of the limits are listed in Table. 3.1.

The cameras used for this design are FullHD USB webcams (Model no: Logicool C615) that can output 1080p @ 30 FPS, and are placed at

<sup>1</sup>The 3D design of the head, and working mechanism was done by Charith Fernando and Youichi Kamiyama.



Table 3.1: Head joints angle limits.

<b>Joint</b>	<b>Minimum</b>	<b>Maximum</b>
<b>Pan</b>	-70°	70°
<b>Tilt</b>	-50°	50°
<b>Roll</b>	-40°	40°

6.5cm IPD [35] to provide stereovision feedback. The diagonal field of view (FoV) for those cameras is 74°. For the depth array, a short-range depth sensor (Model no:Primesense Carmen 1.09) was used due to its small size, and light weight compared with other sensors such as Microsoft Kinect. The effective range for the sensor is 35cm to 140cm, and can output up to 640x480 depth map.

### 3.2.2 **Embeded System**

The internal control unit of the robot that handles network connection, camera processing, head control, and robot control is Intel NUC. With a built in Intel Core i5, USB2.0 x2 ports and 1 USB3.0 x1, it was the best option to adapt in the implementation. The high computational power is required for encoding and decoding image streams from the robot images, and from the user side. The robot utilizes 7 usb ports in total which are as follows:

- Stereo camera uses 2 ports.
- Structure depth sensor uses 1 port.
- Robot head and robot base use 2 ports.
- Usb audio interface for audio input/output uses 1 port.
- Wireless mouse and keyboard uses 1 port.

so several usb hubs were distributed among nuc ports. Figure 3.5 illustrates port mapping for each used device.

### 3.2. Telexistence Robot Implementation.

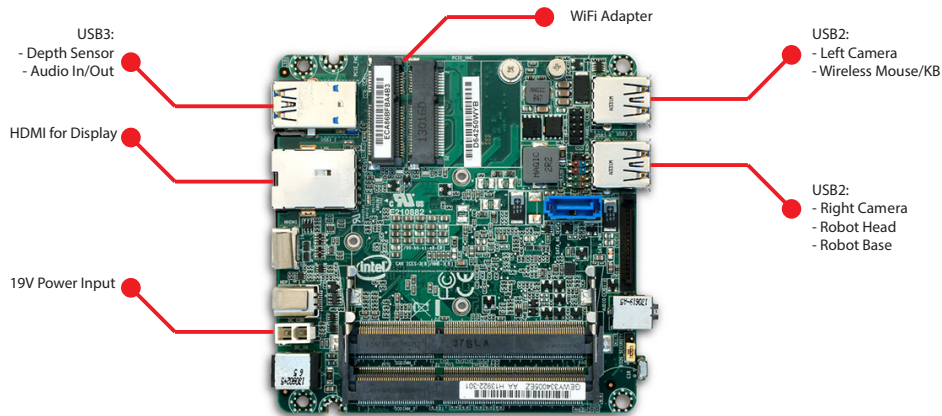


Figure 3.5: Intel NUC board with the corresponding connections

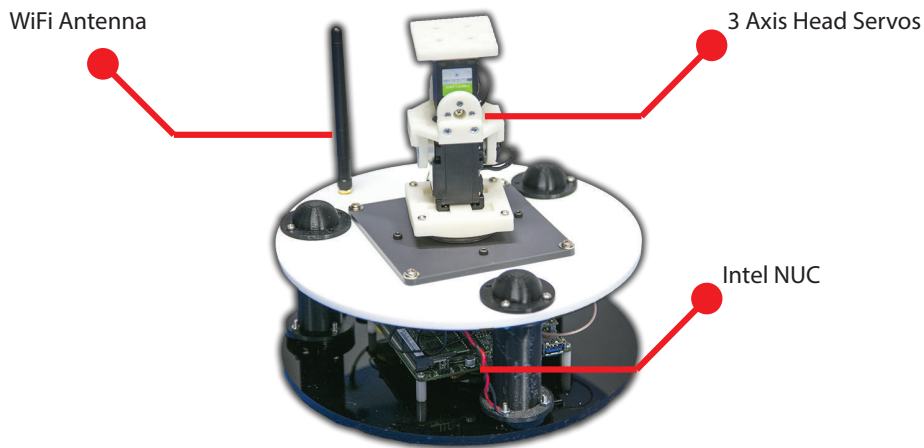


Figure 3.6: Placement of the processing system in correspondence to head

The processing system was embedded inside head module, the advantage of making it a stand alone version that can be mounted on different robots or vehicles. Figure 3.6 shows the final setup of the head module. Intel Nuc is placed under head servos, and all connections are made internally.

#### 3.2.3 Vision Correction

In the design of the head, it was tended to have a wide field of view (FoV) from the robot side for two reasons:

### 3.2. Telexistence Robot Implementation.

---

1. Matching the wide FoV in the HMD ( $110^\circ$ ) that the user wear as much as possible.
2. Provide peripheral vision in the remote place.

In order to get a wider FoV from the robot side, and matching it with HMD FoV, a custom wide lens was added to each camera, the result diagonal FoV is  $100^\circ$ . Figure 3.7 shows the used camera and three types of lenses that has been tested to obtain optimal view and image quality.

In lens selection, usually there is a trade off between the amount of FoV to acquire, and with the chromatic aberration (CA) and barrel distortions. chromatic aberration correction requires a previous knowledge of lens elements and specifically its focal length of each element in order to correct. And regarding barrel type of distortion, the process usually much easier than in CA. A calibration step is applied on the lens in order to obtain radial and tangential factors of the lens. Typically, a chessboard image is used in the process. A very good reference for applying this type of calibration is using OpenCV [36]. In order to apply the correction on the distorted images, using these obtained factors, a real-time shader was applied on camera's images. For details about this shader, please check in appendix appendix A.2.1.

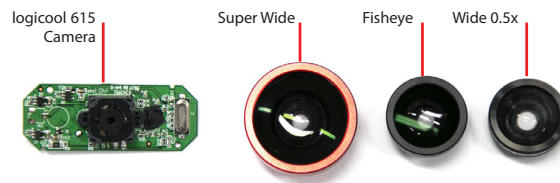


Figure 3.7: Webcam used in the head implementation, and the three tested types of lens

### 3.2.4 Mobility Design

One of the important elements in designing the robot, is to provide a reliable motion and movement in the remote place. The robot should be suitable to move around in public spaces, as well as indoors. Thus the design of the robot movement system should be optimized for these conditions, using a small and lightweight platform, as well as being safe to move with minimum collisions with people and objects in the other place.

To satesify the previous specifications, two different implementations using two types of movable platforms were done, each with its advantages/disadvantages. The specifications of each platform are listed in Table. 3.2. The main considerations for selecting the robot were: The size, weight, and embeded sensors. Speed also play a role in the selection, however its not a main factor. The first version of this system was done using Omni

Table 3.2: Specifications of two mobile platforms used in the system

Platform	Radius	Height	Weight	Speed	Sensors
<b>Omni Wheel</b>	560cm	330cm	20kg	45m/min	None
<b>Roomba 550</b>	270cm	92cm	3.6kg	30m/min	6 IR + 2 Bump

Wheel platform. The omni wheel provides motion on 2 Axis (front/back and side) plus panning rotation. In tests we found this platform has a very large momentum, which in practice can cause damage when it collide with objects or, in worst case with people. Also it was not portable enough due to its weight and size. In the second version, Roomba platform was used instead. The Roomba can provide motion on 1 Axis only (front/back) plus panning rotation. Though this platform has more limits in terms of the payload it can handle and motion, it was sufficient for our application. The two systems can be seen side by side in Figure. 3.8.



Figure 3.8: The two implemented versions using Omni Wheel and Roomba

### **Collision Avoidance**

As it was observed in the initial design of this system using Omni Wheel platform, the user was not quite aware of what is surrounding the robot at low level. So participants sometimes crashes into small boxes or low tables, and sometimes to people feet. To avoid that, a collision avoidance protection level was implemented in the robot level. So even if the user tried to crash into a wall or object, the robot will stop moving in that direction.

Though iRobot Roomba has collision avoidance algorithm, it can not be used in this implementation. The reason is the robot must run on Passive Mode in order to run its internal functions, that is collision avoidance is one of those internal functions. For our purpose, the robot must run on

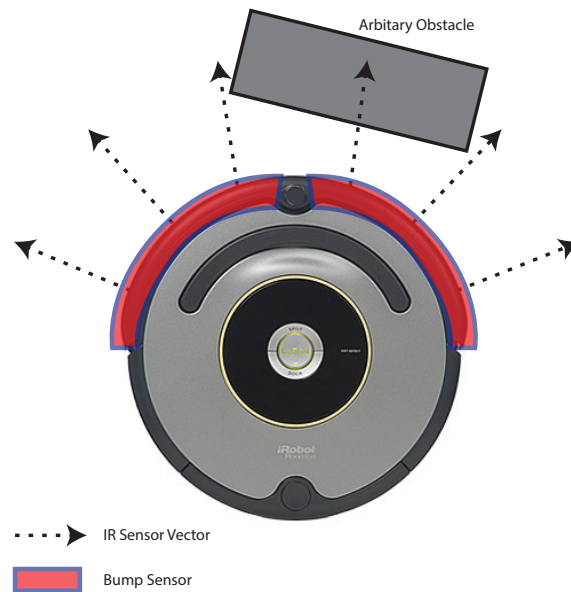


Figure 3.9: Sensors used for collision avoidance.

Full Mode, when the robot is in Full Mode the following happen:

- Grant a complete control over Roomba.
- All safety related conditions are turned off, such as wheel-drop, internal charge safety features, cliff, and bumps.
- Not capable of charging in this mode.

As can be seen, the Full mode has its advantages and disadvantages<sup>2</sup>. To implement collision avoidance, the embedded IR and bump sensors were used to determine if the robot should slow down or stop. Figure 3.9 illustrates the sensors used for collision avoidance. The IR sensors outputs values in range  $[0-65535]$ , 0 means no obstacle in the way. Bump sensor output is  $[0-1]$ , 0 means not activate. The avoidance algorithm outputs three decisions when moving the robot:

1. Path is clear, no restrictions: Happens when no sensor is triggered.
2. Slow down speed: When the IR sensors are triggered.

---

<sup>2</sup>The robot stays in Full mode even if it is disconnected from the PC, so it is very important to return it to Safe Mode before disconnecting, for the ability to charge.

3. Stop: When the IR sensors are triggered beyond certain threshold, or one of the bump sensors is activated.

The avoidance is only applied on the forward motion, and no restriction on the rotational motion. The robot has the ability to rotate at full speed even if it is in Stop condition, allowing the user to rotate and move away. The reason not to apply restriction on the rotation is ,if the rotation of the robot did not match user's motion, sickness or dizziness might occur to the user.

To calculate the slow down speed using IR sensors, first the angle between each sensor is measured physically, the angle is almost 32°between two sensors. Then a set of six vectors are calculated for each sensor as in Eq. 3.1. Basically on the first component (X) of this vector is needed, because the robot can move only front and back, the other component (Y) can be used if the robot can move sideways (like in Omni Wheel platform). However here will be listed a general implementation for both robots. Then each value of the IR sensors ( $V_i$ ) is normalized to the range [0-1], and using Eq. 3.2 a decision value ( $W$ ) is calculated<sup>3</sup> which determine if slowing down should be applied or not. For this, an trial and error threshold values were used, in test we found 0.2 is a good threshold that avoid any noise on the IR sensors. Eq. 3.3 calculates the actual force ( $F$ ) that should be applied on the input speed to slow it down if ( $W$ ) was triggered.

$$\overrightarrow{Sensor}_i = [\sin(170 - i\theta), \cos(170 - i\theta)] \quad (3.1)$$

$$W = \sum_{i=1}^6 \overrightarrow{Sensor}_i \sqrt{V_i}/6 \quad (3.2)$$

$$\overrightarrow{F} = \sum_{i=1}^6 \overrightarrow{Sensor}_i V_i/6 \quad (3.3)$$

---

<sup>3</sup>The square root was used to provide nonlinear slowing down.

For the Stop condition, the robot totally stops if (W) value exceeded a specific threshold, in trial it was set to 0.7 as stopping condition. Another factor to halt the robot if one of the two bumpers has value of 1, it means the robot has physically hit an obstacle.

One more point to highlight is disabling backward motion. The reason is due to lack of sensors behind, and the lack of awareness to the user when he tend to move backward. The user have to turn 180°in order to do so.

The user also gets feedback when one of the previous conditions happen. It is necessary to notify him about the robot status so he can understand why the robot not responding for example.

## Power System

The system runs completely mobile, which means there is no wiring between the system and any external power source. System's power is managed by two sources <sup>4</sup>:

- Internal iRobot battery, which is basically to run the internal motors of the iRobot.
- Two mounted LiPo batteries (14.8V 4s 4600mAh) that handle powering the internal PC and head servos.

In this sytem, three voltage values were required are required to be provided to run. For the NUC part, it requires 19V to run, thus a boost converter (step-up converter) is used, powered by LiPo batteries. A 9V is required to run the head servo motors, so a Step-down converter is used. Finally, a 5V is used to power the display and the speaker, so another step-down converter is used.

The 19V step-up converter is attached with one LiPo, the other LiPo is attached to the 5V and 9V step-down converters. With this setup, we

---

<sup>4</sup>The design and implementation of the power system was conducted by (Charith Fernando).



found the system can run 90 minutes on average before the voltage drops down, basically most of the load is caused by the NUC board. It is highly important not to drain the LiPo batteries below 3.4V/Cell (13.6V in total) in order to avoid damaging the battery.

## 3.3 Operator System

Figure. 3.10 shows what the user typically use to have access to the robot. This setup basically is a combination of HMD (Oculus DK2) to provide visual feedback plus head position and orientation tracking, an extra mounted device for hand capturing and tracking (Leapmotion), and a headset to provide stereo audio to the user (a noise canceling type). All peripherals and softwares related to the user side are running on a single PC. In this setup, the model of the PC used is Dell Precision m4800 (laptop).

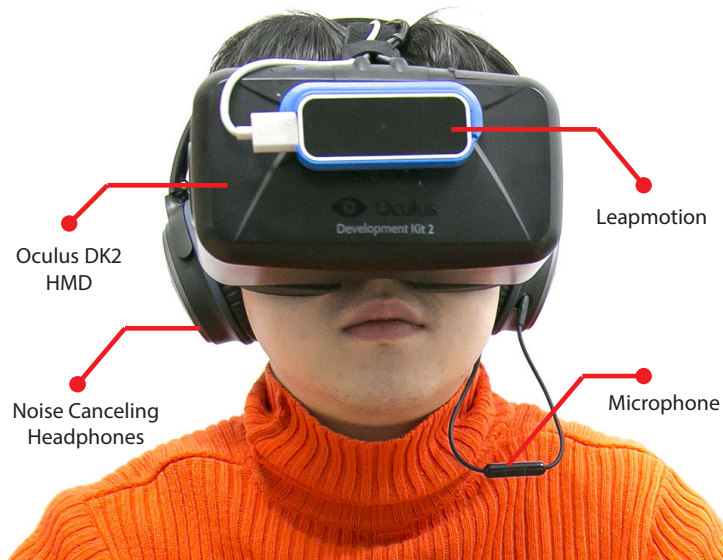


Figure 3.10: Major components for display and tracking.

#### 3.3.1 Tracking System

The main information needed to be tracked from the operator body are the following:

- Head Orientation: To control robot's head rotation.
- Head Position: To control robot's motion.
- Body Visuals: Used for virtual embodying the user in the remote place.
- Hands and fingers position: To calculate touch forces.

Oculus DK2 (the next version from DK1) has the capability to track head orientation and position in the 3D space. For the head rotation, an integrated motion sensor inside the HMD outputs a quaternion representing the 3 rotation axis: Pan, Tilt, and Roll. Only the Pan value (Yaw) needs to be calibrated to determine the front value (where the HMD is facing), usually is done after the user wears the HMD. For position tracking, an external IR camera (part of DK2) is used to capture head position in 3D space, outputting 3 axis values (X,Y,Z).

#### Control the Robot

To drive the robot's head rotation, the process is straight forward: sending the 3 euler angles to the robot. However, due to conversion from quaternion to euler, a gimbal lock <sup>5</sup> happens.

The robot motion is controlled using head motion: Rotation motion along Pan axis, and translational movement along X and Y axis. Figure. 3.11 shows the motion vectors for the translational and rotational movements.

The idea of the control is to mimic how our body moves when we tend to rotate or walk forward. Since the operation is done while being sit, only

---

<sup>5</sup>Gimbal lock:losing one degree of freedom due to casting quaternion (tuple of 4 values) into euler angles (tuple of 3 values). So in cases when two rotation axis overlap, it becomes unknown how to resolve the angles.

### 3.3. Operator System

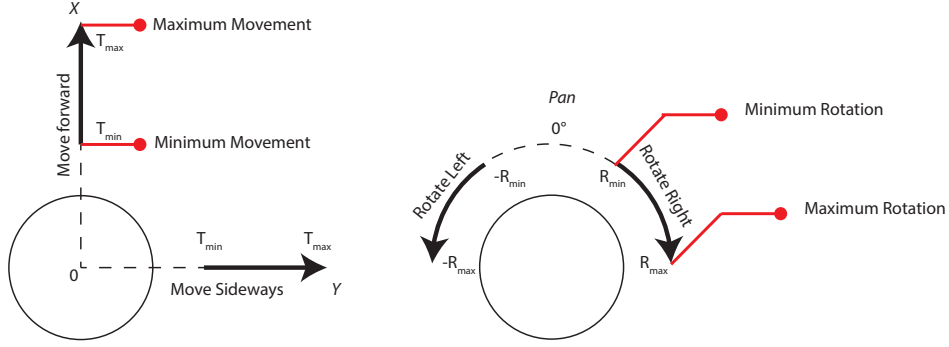


Figure 3.11: Illustration of HMD motion based control.

relative movement is taken to consideration. When the user start looking around, the robot head is being controlled by that motion. However when he exceed a certain threshold  $R_{min}$ , the robot start rotating according to the direction. Eq. 3.4 calculates the rotation speed of the robot.

$$Rotation = sgn(\Theta)((\Theta - R_{min})/(R_{max} - R_{min}))^2 \quad (3.4)$$

Using the previous mechanism, the movement speed is determined based on the head offset from the original position. When the head position on one of the two axis (X,Y) exceeds a minimum threshold  $T_{min}$ , the speed of the robot follows accordingly. Eq. 3.5 calculates motion speed of the robot.

$$\begin{aligned} Speed_x &= sgn(x)((x - T_{min})/(T_{max} - T_{min}))^2 \\ Speed_y &= sgn(y)((y - T_{min})/(T_{max} - T_{min}))^2 \end{aligned} \quad (3.5)$$

By testing the system with several users, we found the nonlinear motion was more natural than linear. The parameters ( $T_{min}, T_{max}, R_{min}, R_{max}$ ) are also been tuned via user tests. The smaller the min values are, the more likely the robot will start moving when the user slightly move. The higher the values are, the more the user need to move in order actuate robot's motion.

#### Hand Tracking

In an early stage of this research, the tracking of user hands was done via 5DT data gloves. And the kinematics of his arms was tracked via Natural point OptiTrack system. The setup was relatively complex (the amount of wearables the user need to put) and the results of tracking were poor.

In the second iteration of this system, the complexity of the system was tremendously reduced by switching from bending sensor type of gloves into image based tracking using Leapmotion sensor. This sensor uses two stereo super-wide FoV (150°) IR cameras that captures hands images, and estimates the posture of the hands and fingers based on that. The original setup for this sensor was as a hand gesture input device placed on table top. This sensor eventually emerged into virtual environments as a hand tracking device. The sensor outputs hand joints positions and orientations in the 3D space relatively to sensor position. Also it outputs the captured IR images for application specific use. Figure. 3.12 visualize the captured IR images with the respective hands skeleton aligned with them. In this

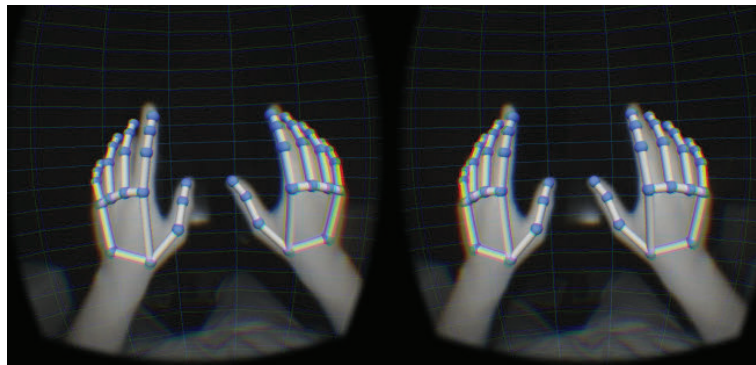


Figure 3.12: The captured IR images of the hands with joint information for each finger

research, Leapmotion sensor was placed in the front of the HMD in order to cover eyes' visual area, and to always capture hands from the eyes point of view. As mentioned before, the captured joint frames of the fingers and hands are all in the Leapmotion space and not in the absolute 3D world

space. All positions and orientations must be converted into the 3D world space coordinates, and to do so leapmotion coordinates must be identified. Luckily in applications where the Leap is mounted on the HMD, the location of the Leapmotion is always relative to the location of the HMD. And since we have already the absolute frame of the HMD (position and orientation tracking), then the Leapmotion frame is also known and derived from HMD frame. So if the HMD frame is represented as a Homogeneous Matrix (4x4) namely  $M_H^0$  which converts any point or frame from HMD frame (H) into the world frame (0). Then the calculation of hand joints are easily done using Eq. 3.6.

$$J_L^0 = M_H^0 J_H \quad (3.6)$$

The fingers joint are now converted into the world space and can be used to respond to user's interactions in the world coordinates, as well as for haptic estimation as described in a later section.

#### 3.3.2 Virtual Embodied Arms

Two implementations for virtual hand embodiment were done. Each method uses different approach to capture and render the visual feedback of user's body on the remote vision. The two methods are: Virtual 3D Embodied hands, and Image-based Embodied hands.

##### Virtual 3D Embodied Hands

The first method to provide body visuals is using computer generated (CG) body representation. In this technique, we used the tracked user body information of the arms and hands (joints position and orientation) and mapped them into 3D virtual arms. Figure 3.13 (a) shows the resulted arms being superimposed into the remote vision. Also, in this method, finger tips position are already calculated from the tracking algorithm, that allows us to use haptic estimation algorithm (discussed in 3.3.3). Figure

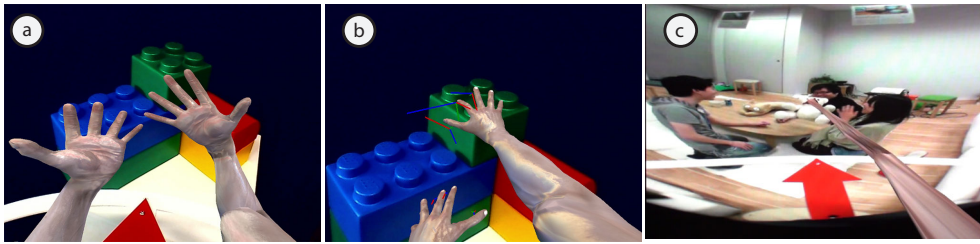


Figure 3.13: (a) Virtual 3D hands superimposed on remote environment. (b) Using the hands to touch objects. (c) Superhuman ability, extending the arms.

3.13 (b) shows the interaction between the virtual 3D hands with the remote physical objects.

This method exposes the possibilities to alternate some properties of human's body. For example, user can experience being inside a different body, machine body or even an animal body. Also, expanding the functions of the body, for example extending user's arms to reach far beyond what it can physically do as shown in Figure 3.13 (c). This technique was previously proposed in virtual environments interactions [37].

However, in this technique the user does not perceive the body as his own body due to the visual differences, and the uncanny look of the CG body.

#### Image-based Embodied Hands

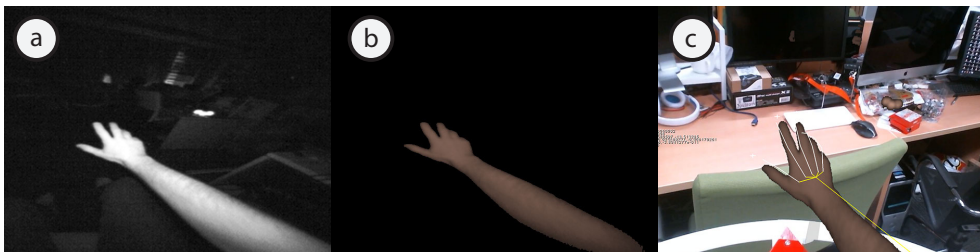


Figure 3.14: (a) Captured hands from Leapmotion (b) Processed image, remove background (c) Final superimposed hands

The other approach to provide hands visuals to the user is by capturing them using a camera and process them to isolate them from background.

Previous techniques has been used to achieve this in Virtual Environments such in [38] using chroma-keying technique, that is by placing a unique color as the background and then removing this color in the software. Using chroma-keying technique, a very high quality visuals are resulted for the body, however this required special setup for the user's environment.

Here we propose using IR camera that is embedded inside the hand tracking sensor (Leapmotion). The hands are segmented from the background based on the intensity of the reflected IR to the camera. Figure 3.14 (a) shows the captured hands from the IR sensor. The removal of the background noise is shown in Figure 3.14 (b). And the final corrected hands scale and position are shown in Figure 3.14 (c) with the integration in the remote place.

#### **3.3.3 Haptic Feedback System**

Though the system does not have any physical haptic sensors in the robot side, a pseudo haptic estimation method was implemented in this system. The method uses a structure depth sensor in the robot side to create a representation of the remote place. The depth values are sent to user side to estimate touch forces. The method works on two main steps: Reconstruction the remote environment surface information using depth data, and estimating touch and stroke forces using fingers information and the calculated surface data.

##### **Depth-based Environment Construction**

The goal of this step is reproduce the remote environment from robot side point of view. The reconstruction is derived from the depth data, which is in this implementation a 2D array of 320x240 11bit depth values. Several challenges remain when using raw depth data from the sensor. Problems such as resulting noise from the sensor, and missing values or 'holes' that no

structured light depth reading was possible to retrieve. This can be the result of certain materials in the environment does not reflect IR light back to the sensor. To solve such issues, incremental methods create spatial representation of the environment using series of captured depth-maps [39]. Such methods are effective for robotic investigation in unknown environments, and for AR based applications. However such systems are optimized to construct static scenes and non-movable objects such as people passing by the camera which is a common case for telexistence applications. For our purpose, we are interested in realtime reconstruction, thus we calculate the scene surface per-frame with no temporal information.

The method is aimed to be used in real-time, thus we avoid using polygonal representation of the 3D scene due to the complexity, content size, and transmission. The calculations are done on the local side, and only a surface map (or normal-map) is generated from the depth-map. Using this representation it is possible to generate not only pushing forces, but also shearing forces along finger's surface.

The captured depth is processed and clean up before being used to generate scene surface. To do so, first a filling strategy for the missing depth values, then a smoothing filter is applied on the resulted depth-map to remove the presented noise while preserving the hard edges of scene objects. To recover the gaps of the depth-map, a statistical search model is applied for each missing pixel. The model creates a sliding window and picks the value highest frequent depth in this window.

Scene surface normal-map is derived from depth-map values. Each pixel of the normal-map is a 3D vector representing surface direction at that point. Normal vectors are calculating using the cross product of the derivatives along X and Y space of the depth values using Eq.3.7.  $\partial U$  and  $\partial V$  are the



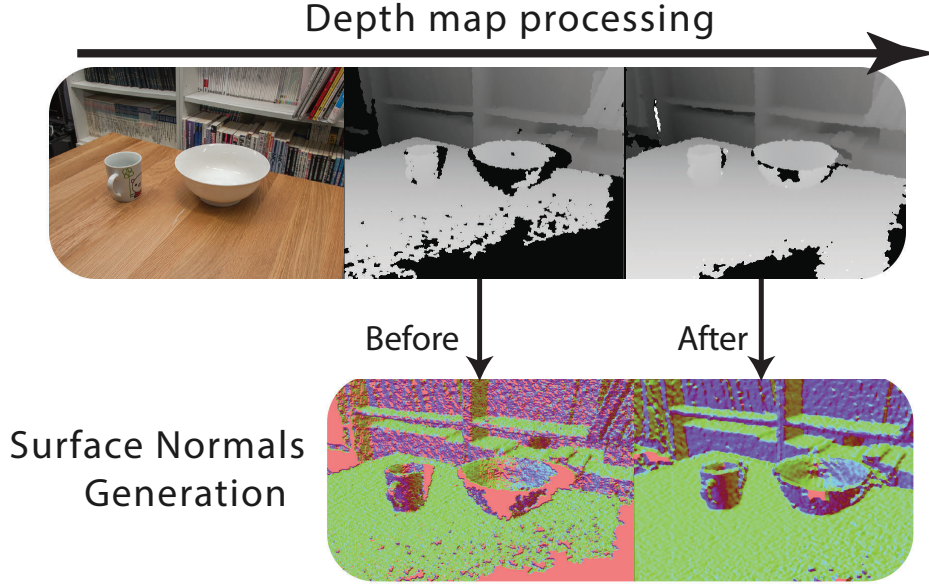


Figure 3.15: Processing the captured depth map to generate surface map derivative vectors along X and Y axis respectively.

$$\begin{aligned}
 \vec{N}_{xy} &= \left\| \vec{\partial U} \times \vec{\partial V} \right\| \\
 \vec{\partial U} &= [\partial x, Depth, y] \\
 \vec{\partial V} &= [x, Depth, \partial y]
 \end{aligned} \tag{3.7}$$

However, the noise from the depth sensor become obvious on the normal-map, thus a smoothing filter is applied before calculating the normal. Applying a Gaussian filter on the depth values which works as a low pass filter in the image space. However it does not take into consideration the hard edges of the objects in the image. Thus a bilateral filter is used [40] and applied on the depth-map. The entire process to generate surface normals is shown in Figure. 3.15.

#### Haptic Estimation on user's fingers

To calculate the touch forces with the environment, user's fingers position in the 3D world space are used as well as the constructed surface map of the remote environment. The position of the finger is projected into eye space in order to match the same coordinates of the surface map points. The Z component of each projected finger represents the depth from the eye to finger tip. The touch force on the finger tip is calculated using Hooke's law, the difference between finger depth value and the remote place depth is taken as the input parameter for the function. Eq. 3.8 outlines the calculation of the touch forces. And Figure. 3.16 shows the process of estimating the touch forces using depth map and finger position in the 3D space.

$$x = Z - D$$
$$f(x) = \begin{cases} kx, & \text{if } x > 0 \\ 0, & \text{otherwise} \end{cases} \quad (3.8)$$

#### Haptic Display Module

The developed haptic display is based on Gravity Grabber [41]. The display renders pressure forces and shearing forces on the finger tips. However in this design, only pressure forces are presented due to size constraints. Pressure forces can be displayed using one motor and belt, but to present shearing forces, two motors are needed per finger. The reason the size is important to consider is because of the tracking system (Leapmotion) can not determine finger posture if the shape of the finger was changed a lot due to wearable devices. Also another consideration in designing a wearable device for Leapmotion is the ability to reflect IR light back. So in this implementation, a layer of retro-reflective material was added on the display.

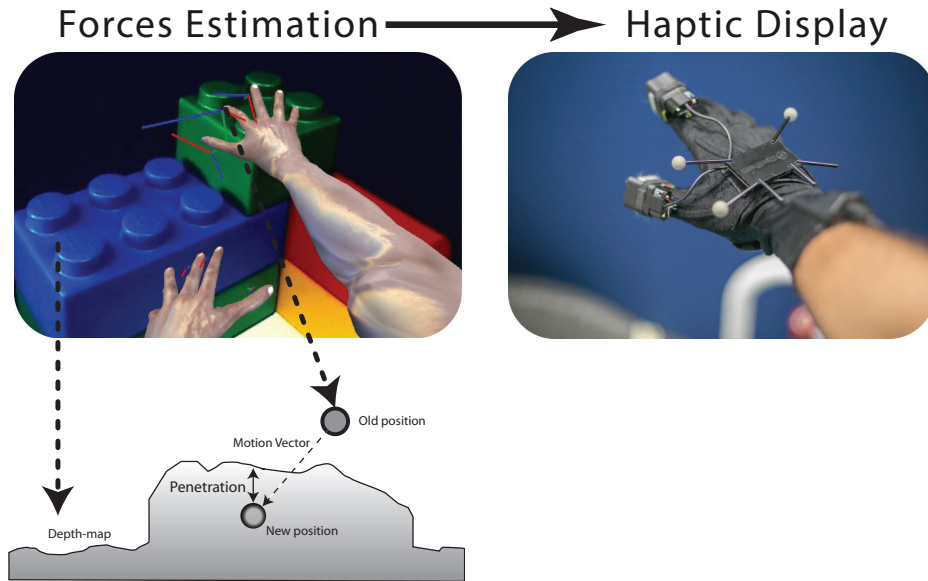


Figure 3.16: Estimating touch and stroke forces on user fingers using SM

The control box for the display has been dramatically reduced in size from

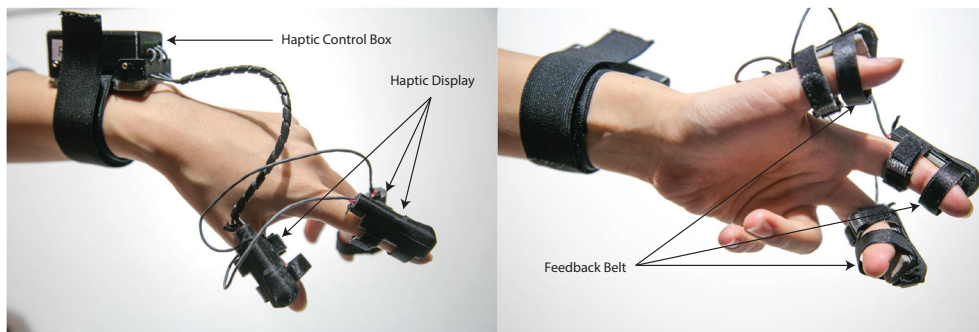


Figure 3.17: Touch feedback display

the original display. The display is completely wearable as can be seen in Figure. 3.17. The processing board used in the display is mbed LPC1768, and the motor drive is Texas Instruments DRV8833 that can handle two motors. For each hand, three fingers (Thumb, Index, Middle) are each mounted with one display, and the control box is mounted on the wrist. The control box is interfaced with the PC using serialport connection over USB. Also a 5V power is supplied to the motor drives. The control box, and haptic displays parts are all custom 3D manufactured parts.

The calculated forces are sent to the haptic daemon over udp, generally both are running on the same pc (haptic calculator, and haptic display daemon). For touch forces only type of feedback (no shearing), a tuple of 6 values are sent, each value is mapped to finger (Right: Thumb,Index, Middle),(Left: Thumb,Index, Middle). These values are used to drive each motor of the display.

## 3.4 Communication System

The robot and user are designed to be separated and not using any direct connection between each other except via a wireless network (Local Area Network/Internet). Selecting the communication method is important to provide real-time contents delivery between each other. The type of contents which both systems (user/robot) deliver to each other are as follows:

1. Video Stream: Both the user and robot send to each other real-time video stream, with a quality that can reach 1920x1080 resolution @ 30 Frames (especially from robot side).
2. Audio Stream: Also both systems delivers an audio feedback from each other, the robot should provide multi-channel audio to the user with quality 22000Hz.
3. Depth Data: The robot scans the remote environment and sends back a 2D array of depth data with size 320x240 11bits of resolution.
4. Control Signals: Both systems have a communication channel for state and robot control.

### Depth and Control signals Communication

The Video/Audio streams are separated on a different channel from the other data streams to ensure an isolation between both. The depth data is

### 3.4. Communication System

compressed using ZIP format and sent periodically (50ms) to the user side.

The control signals represent the state of the system, control motion to the robot, requests from user to robot, and indications from the robot to user. The signals are sent using XML format, this helps to extend the protocol in the future. Table. 3.3 lists the used control signals in the current implementation. The table divides the signals from the user and from the robot.

Table 3.3: Control signals used between user and robot

	Signal Name	Description
<b>From User</b>	Connect	Notify the robot that a user has connected
	Disconnect	Notify the robot that a user is disconnected
	Speed	Control movement speed of the robot
	Rotation	Control rotation speed of the robot
	HeadRotation	Control robot's head orientation
	HeadPosition	Control robot's head position (if applicable)
	CalibrateDone	Notify the robot the user has callibrated
<b>From Robot</b>	DepthSize	Depth image dimensions
	IsStereo	Is robot supports stereo vision
	CameraConfig	Configuration for the camera (for correction)
	IRSensor	IR Sensors values (for collision avoidance)
	BumpSensor	Bump Sensors values (for collision avoidance)
	BatteryLevel	Robot's battery level
	DepthSize	Depth image dimensions
	DepthData	Depth stream
ReportMessage	Notify the user if a problem happened	

## Video/Audio Communication

The video and audio feedback system is implemented for both sides: The user and robot. Providing a low latency feedback for both ends is a critical point in this system that has been carefully done. The remote side sends to user stereo images of the left and right eyes of the robot. The user only provides a single video stream for user's top body to the robot. The

### 3.5. User Interface

---

requirements for the robot side video quality and latency is more critical than from the user side. Table 3.4 summarizes the required video quality for each system.

Table 3.4: Video quality requirements for the robot and user side

	Camera	Min Resolution	Min Bitrate
<b>Robot</b>	2	1280x720	2000 Bps
<b>User</b>	1	640x480	500 Bps

To realize the video streaming, H264 format was used for encoding and decoding the video stream. This format is considered a standard in video media streaming over network, and provides wide tuning parameters to achieve the desired performance. For both systems, the parameters are tuned for the minimum latency, so the encoder disables any buffered frame that can cause latency (However this tuning affects the image quality).

For audio streaming, also the robot sends 2 audio streams for the left and right ear. The user sends only 1 audio stream to the robot speaker. For audio streaming, AMR format was used for encoding and decoding. For more details about the pipelines used and the selected parameters used for the video and audio streaming are listed in the appendix A.3.

## 3.5 User Interface

The amount of UI contents the user see while operating should be as minimal as possible to avoid any disconnection from the remote presence experience. Only meaningful contents were presented, and it appears only when an important event or feedback the user should know. The elements of the UI were added after observing user's operation of the system. We found that the user sometimes is not aware of the speed of the robot, or if the robot is actually rotating or not. This can be an issue of the limited peripheral vision of the camera and HMD. The speed of the motion appears

as a small arrow that moves in the direction of robot movement Figure 3.18 (a).

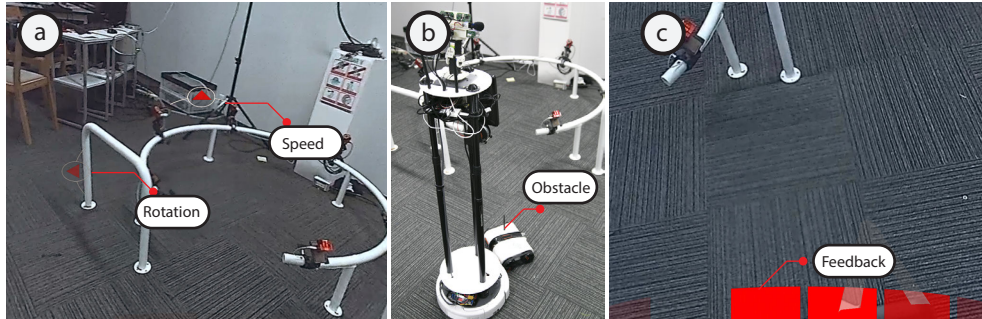


Figure 3.18: User interface depending on the scenario.

The UI also provides feedback when the robot collides with obstacles. As discussed in 3.2.4, the robot detects any nearby obstacles, and it stops in case of collision. Sometimes the user is not aware of the collision and would be confused of the reason the robot is not responding. To solve this issue, an indicator appears in the bottom of the screen that shows the location of collision. Figure 3.18 (b) shows the robot colliding with a small toy the user could not see, and in Figure 3.18 (c) a feedback appears in red color allowing the user to understand the location of the obstacle.

## 3.6 Technical Evaluation

Evaluating the speed of transmission in any telexistence system is necessary to know before conducting user evaluation for it. The two main concerns regarding the latency are: body perception latency, and Remote visual feedback latency. Those two factors are necessary to minimize to an acceptable level. “Acceptable level” has two cases, one for each of the previous:

- The first factor related to body perception affects user’s kinaesthesia, which is the awareness of his body position in respect to the motion.

### 3.6. Technical Evaluation

---

When user experience latency towards the presented body, the level of presence is reduced accordingly.

- The second factor related to visual feedback latency mainly affects the operation efficiency of the robot (navigating for example), and also the motion sickness the user would experience if the visual feedback did not match his motion head motion.

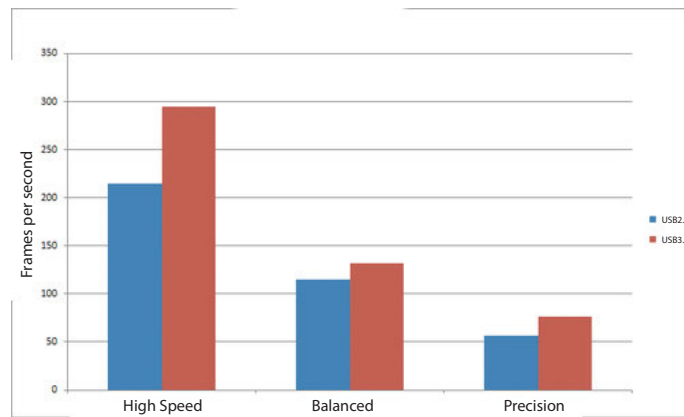


Figure 3.19: Leapmotion framerate comparison while running on different modes.

For body perception related latency, the processing of body and hands visuals are entirely done locally, no network is involved in this process. The speed of capturing the hands images, filtering them and rendering them does not exceed 30ms. Also, as has been reported previously by the developers of Leapmotion, the frame rate of image output from the devices is affected by the running mode the device is using (High-speed, Balanced, Precision) and the connection with the PC. Figure. 3.19 shows the effect of changing the mode of the device and the outputted framerate.

### Visual Feedback latency

The video feedback coming for the robot has been tested using two video resolutions: HD Resolution 1280x720 at 30 FPS and FullHD Resolution



### 3.6. Technical Evaluation

1920x1080 at 30FPS. And for each resolution, several bitrates ranging from 2000 Bytes/sec to 7000 Bytes/sec has been tested.

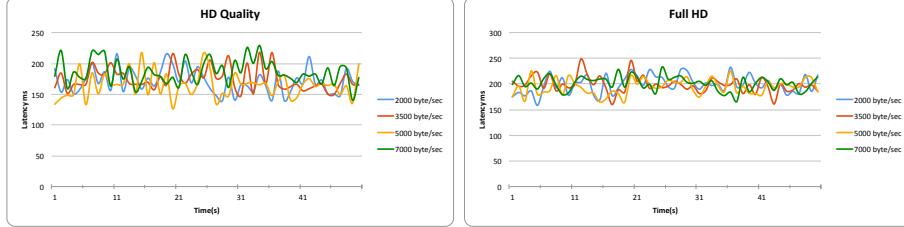


Figure 3.20: Video stream latency over time

The goal of this evaluation is to understand the effect of encoding and compression (CPU utilization) versus network delay. The less the bitrate is, the higher the compression is and less packet and frame loss. When the bitrate goes high, the required compression become less and the network delay goes higher due to packet loss. Figure. 3.20 shows time vs latency graph for several captured frames over 1 minute. And Table. 3.5 provides the mean and standard deviation of the captured samples latency. The video stream evaluation shows that streaming at 5000 Bps provides a

Table 3.5: The average latency in millisecc using two video streaming resolutions and several bitrates

Resolution	2000Bps	3500Bps	5000Bps	7000Bps
<b>HD</b>	171 ± 20 ms	174 ± 18 ms	164 ± 21 ms	186 ± 20 ms
<b>FullHD</b>	199 ± 17 ms	198 ± 16 ms	193 ± 15 ms	201 ± 13 ms

balance between the compression and bandwidth, with 164ms/HD 193ms/- FullHD are the least latency compared with the other cases. Though the minimum latency that has been achieved is 164 and is considered as high latency compared with a direct link system (latency of rank 50ms), it is acceptable for network based environment. Still there are several technical challenges to reduce this delay: Capturing latency from the USB2.0 cameras (30ms), encoding the image frame, compression the encoded stream, streaming it over the network, uncompressing and decoding the stream, display to screen.

# Chapter 4

## Evaluation and User Study

Previous chapter has provided the details of the implementation process for the Ubiquitous Virtual Embodied Telexistence system that is proposed. We showed the system parts and the intended experience through it. We also highlighted previously the importance of body presence when telecommunicating through a telexistence system. In the final implemented system, the user experiences his own body being replicated in another place, also the user is capable to move in the remote place using his body motion. In this chapter, we evaluate the system and the level of presence when using this system. The user study analyses how would the first time users tend to operate the system. Also we observed their reaction when they see their body visuals being presented remotely. These observations helps for behavior analysis when operating.

### 4.1 Evaluation of Presence

We conducted user studies to verify our proposed concept and the realized system. The goal is to study if the virtual embodiment concept gives the user sense of being in another place. Also we measure how effective the movement approach we proposed. Our hypothesis is the addition of the virtual embodiment (the addition of virtual hands) increases the sense of presence for the user, and makes him more engaged in the other place. During the experiment, we observed user's behavior when trying to move the robot, and when the hands were enabled and disabled.

Prior to conduct this evaluation, several pilots studies were done to tune the performance of the system, and the motion parameters for the robot. After finding out the best parameters, the participants were asked for the objective evaluation.

#### 4.1.1 Participants

For this study, 10 first time users joined the experiment (8 Males and 2 Females). The age range for the participants is 22 to 32 years (Mean: 26 - SD: 3.6). Basic information were collected from each participant at the beginning, 3 out of 10 reported they used to have stereo sickness in general (unable to see 3D contents). Figure 4.1 lists how frequent the participants used HMD before. This type of analysis was needed to understand how different users will react to the HMD and stereo vision.

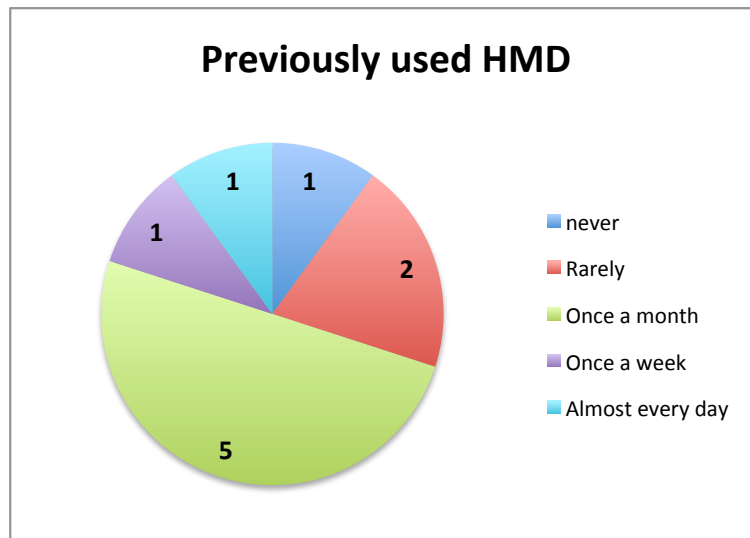


Figure 4.1: Number of participants used HMD before.

#### 4.1.2 Environment Setup

The experiment was conducted in a living room setup. The room was divided into two parts, joined via a curve. Both parts were well lit, and

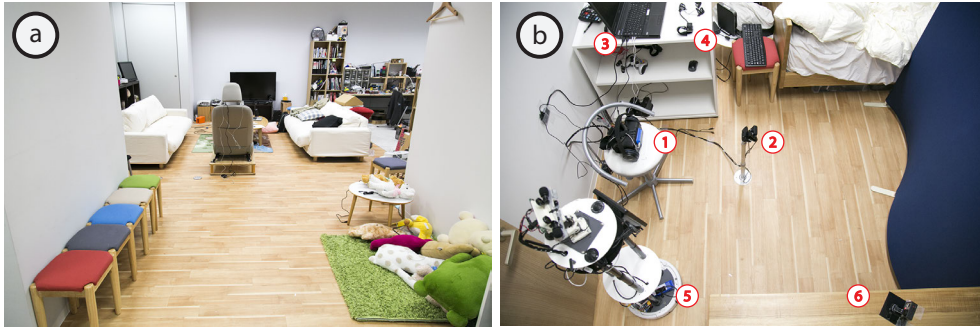


Figure 4.2: Experiment environment setup (a) Robot room. (b) User room.

contained several colorful objects, tables and chairs as can be seen in Figure 4.2 (a). This setup reflects one of the intended applications of this system, adapting telepresence platforms into our daily life environments. Several participants did not have a previous knowledge about the layout of the rooms.

The experiment were conducted individually for the participants. The participant were situated in an isolated small room that contained the tools for controlling the robot. Figure 4.2 (b) shows the setup of the user's room, the labeled items are as follow:

1. The chair were the user sits and the HMD.
2. Head tracking camera, and webcam to capture user's face and send it to the robot screen.
3. The control PC.
4. Two haptic displays for user hands.
5. The robot (when experiment begins, it is moved to the other room).
6. Experiment recording camera, used for behavior analysis.

### 4.1.3 Experiment Procedure

The experiment is divided into 3 phases and on average it last for 20 minutes per participant. Each phase, the user experience the system with

different conditions, and it lasts for 4 minute in average. At the beginning of the experiment, in between each phase, and at the end of it the user is asked set of questions to evaluate his current state and how each experiment affects his level of presence in the other environment. Several questions were selected from a previous study about presence in virtual environments [18].

The three phases of the experiment are as follow:

1. Phase 1: Testing the mobility of the system for the first time. Observing users reaction when being connected and asked to figure out how to move.
2. Phase 2: Testing how the user react when using this system while no hands visuals or haptic feedback are provided.
3. Phase 3: Same scenario Phase 2, but the hands visuals and haptic feedback are being added.

To avoid any bias in the experiment regarding measuring the virtual embodiment (Phase 2 and Phase 3), the participants were split into two groups (A,B). Each group has different order for Phase 2 and Phase 3 of the experiment. Group A starts with Phase 2, and then Phase 3. Group B is Phase 3 then Phase 2.

At the beginning before starting Phase 1, set of general questions are asked to the user to know about his background in regards to Virtual Reality (HMD usage, stereo sickness, visual impairment). The second set of questions are to measure his physical, mental and emotional state before conducting the experiment. Table 4.1 lists the questions been asked with answer range from 1 to 6. At the beginning, the answers for being Dizzy or Disturbed were the same which reflects a very good condition for the users.

Table 4.1: Set of questions before starting experiment

Initial Stage				
Question	Answer Range	Average	STDEV	
How well do you feel now?	1-6	4.9	0.32	
How physically fit do you feel now?	1-6	5.1	0.57	
Do you feel dizzy now?	1-6	1.7	0.67	
Do you feel disturbed now?	1-6	1.7	0.67	

Answer Range: 1: Minimum. 6: Maximum.

### Evaluation Phase 1 - Motion Control

Before starting Phase 1, the user get a breif idea of what the system is, and what the experiemnt is about. No instructions about how to move the robot are provided. After setting up the user with the HMD and haptic gloves, the user is connected to the robot, all the communication with the user from this point is done from the robot side.



Figure 4.3: Phase 1 Experiment, User controlling the robot using his body motion.

The user is asked first to experience his head rotation by looking around. When the user starts looking to the left or right, he realizes the rotation of the robot from the optical flow. Next, the user is asked to start moving forward and follow the examiner. At this point, the user has no previous knowledge of operating the robot, but 9 out of 10 were capable to under-

#### 4.1. Evaluation of Presence

---

stand that using body motion the robot can be controlled. The participants spent almost 2 minutes to adapt robot speed, especially regarding curve motion. Figure 4.3 shows the participant leaning his body to move the robot. Each participant spend about 4 to 5 minutes using the robot, and moving around the room. The goal here is to get them adapted to the system motion and control.

At the end of Phase 1, the users are asked to answer set of questions. Table 4.2 lists up four questions been used to measure how natural the experience was for the user. The initial results of the controlling showed to be high for the participants ( $5.1 \pm 0.74$ ). The head motion control seemed to low compared to the body control, the reason of that is the limits of head movement (refer to Table 3.1), as well as the latency from the video stream (refer to Table 3.5).

Table 4.2: Set of questions after first phase: Controlling the robot

<b>Control Robot</b>				
	<b>Question</b>	<b>Answer Range</b>	<b>Average</b>	<b>STDEV</b>
	Control level in the remote place	1-6	5.1	0.74
	Natural head motion	1-6	4.3	1.06
	Natural body movement	1-6	4.8	0.63
	Presence level	1-6	4.8	0.63

Answer Range: 1: Minimum. 6: Maximum.

#### **Evaluation Phase 2 - No Hands**

After the user had basic knowledge of how to operate and navigate the robot, he is asked to repeat the experiement but this time the user is encouraged to use his body and hands for communication. The user spends a period of time 5 minutes in average talking with other people in the other room.

For Group A, this phase comes directly after Phase1, and for Group B it comes after Phase 3. After the end of this phase, the user is asked to

#### 4.1. Evaluation of Presence

answer set of questions about the experience he had. Table 4.3 lists the questions with answers divided by the group. A discussion about these results will follow.

Table 4.3: Set of questions for Group A and B after phase 2 - No hands are presented

<b>Group A</b>				
<b>Question</b>	<b>Answer Range</b>	<b>Average</b>	<b>STDEV</b>	
Natural Interaction	1-6	4.6	1.52	
Body presence	1-6	4.4	1.14	
Involvement level	1-6	4.8	0.84	
Disturbed or confused	1-6	3.8	1.78	
<b>Group B</b>				
<b>Question</b>	<b>Answer Range</b>	<b>Average</b>	<b>STDEV</b>	
Natural Interaction	1-6	3	1.41	
Body presence	1-6	4	1.41	
Involvement level	1-6	3.2	1.30	
Disturbed or confused	1-6	4.6	1.14	

Answer Range: 1: Minimum. 6: Maximum.

#### Evaluation Phase 3 - With Hands



Figure 4.4: Phase 3 User experience his hands, and uses them to communicate.

In the third phase of the experiment, the user repeat the same steps in Phase 2, except the hands and haptic feedback is enabled. The behavior of the participants in Group A was different from Group B when the hands



#### 4.1. Evaluation of Presence

were enabled. The participants were more actively using their hands while operating the robot, Figure 4.4 shows one scenario of using the hands to communicate with a remote person.

The same set of questions asked after Phase 2, are also asked in Phase 3. There was a noticeable differences between both sets. Table 4.4 lists the questions with answers divided by the group.

Table 4.4: Set of questions for Group A and B after phase 3 - Hands are presented

<b>Group A</b>				
<b>Question</b>	<b>Answer Range</b>	<b>Average</b>	<b>STDEV</b>	
Natural Interaction	1-6	5.2	0.84	
Body presence	1-6	5.2	0.84	
Involvement level	1-6	5	0.71	
Disturbed or confused	1-6	2.4	1.67	
<b>Group B</b>				
<b>Question</b>	<b>Answer Range</b>	<b>Average</b>	<b>STDEV</b>	
Natural Interaction	1-6	3.6	0.55	
Body presence	1-6	4.4	0.55	
Involvement level	1-6	4.8	0.84	
Disturbed or confused	1-6	2.4	1.52	

Answer Range: 1: Minimum. 6: Maximum.

Comparing the results from Phase 1 and Phase 2 (in Figure 4.5) shows the effect of adding hands to the telexistence system. The results from the second test should be seen relatively from the first test. Group A had higher results for the natural interaction and body presence because the hands were added, also they reported to be less disturbed from the no hands phase. Group B showed less involvement and higher level of confusion because the hands were removed.

#### **End of Experiment**

After finishing the experiment, the user is asked the same set of questions he had at the beginning regarding this physical, mental, and emotional

#### 4.1. Evaluation of Presence

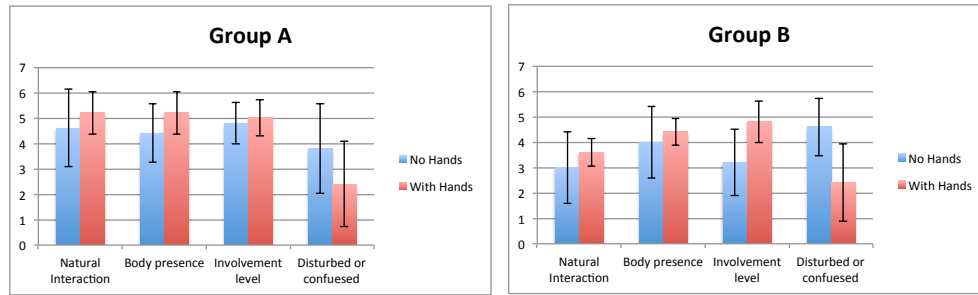


Figure 4.5: Results of the evaluation of Group A and Group B.

state. This is used for comparison with their initial condition, and to understand the usability effect of this system. Another feedback is also provided from the users A about the overall experience and how compelling the system was.

Table 4.5: Set of questions after the end of the experiment

After Experiment				
Question	Answer Range	Average	STDEV	
How well do you feel now?	1-6	4.6	0.97	
How physically fit do you feel now?	1-6	4.7	0.82	
Do you feel dizzy now?	1-6	2.7	1.42	
Do you feel disturbed now?	1-6	2.1	0.99	
Hands visuals compelling	1-6	3.8	0.92	
Hands feedback compelling	1-6	2.9	1.20	
Surrounding sense	1-6	4.7	0.82	
Capable to move freely	1-6	5	0.67	
Sensory feedback compelling	1-6	4.3	0.82	
Involvement level	1-6	4.6	0.70	
Natural Interaction	1-6	4.4	1.07	
Least compelling feedback				70% Touch

Answer Range: 1: Minimum. 6: Maximum.

By plotting the results for participant status before the experience and after the experience, we can see in Figure 4.6 the differences. It should be obvious that the user will be less comfortable using a telecommunication system than the status of being communicating directly. However as can be seen in the comparison, the users did not report significant change in

the physical state and how good they feel after the experiment. However several reported feeling dizziness after finishing. The result can be mainly regarded to the visual latency they experience when rotating their head.

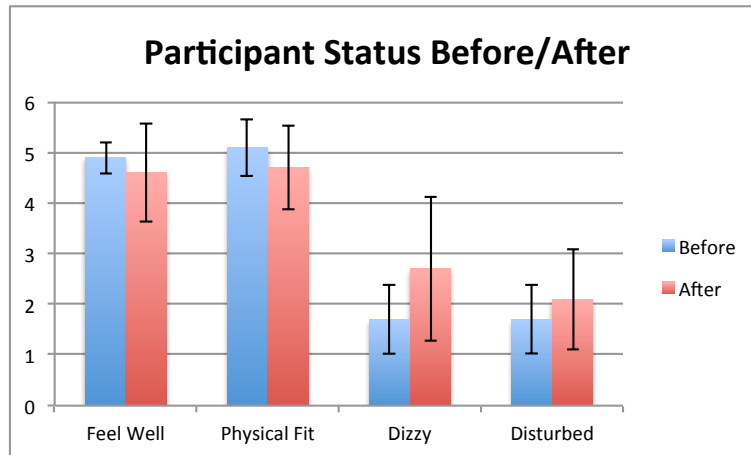


Figure 4.6: Changes of participants state before and after the experiment

#### 4.1.4 Behaviour Analysis

We observed participants' behavior during the experiment using a recording camera. This helped us to understand the amount of movement and effort the user put, as well as to evaluate their expressions when the hands are added and removed. In general, participants did not find it difficult to control robot's movement after finishing the first phase. They managed to figure out how to move after experiencing the head movement.

For Phase2 and Phase3, the behaviour of GroupA was different from GroupB. GroupA were excited after they saw their hands, and found it much more natural the experience when their hands were presented. GroupB in contrast, they were disappointed after removing the hands, even they expressed their feeling as "taking away part of their body" during the test. However, both groups reported that the touch was the least compelling for them. This can be regarded to the knowledge of the feeling of "touch" they expected to experience, and the actual haptic feedback they received when

they used their hands.

## 4.2 Public Subjective Evaluation

Outside the research lab, the system has been presented to a wide group of people in different events and academic conferences. With over 400 person tried this system, we were capable to understand more deeply the usability requirements for this system, and fine tune many parts of this system like the safety requirements, motion control, body visuals and performance.

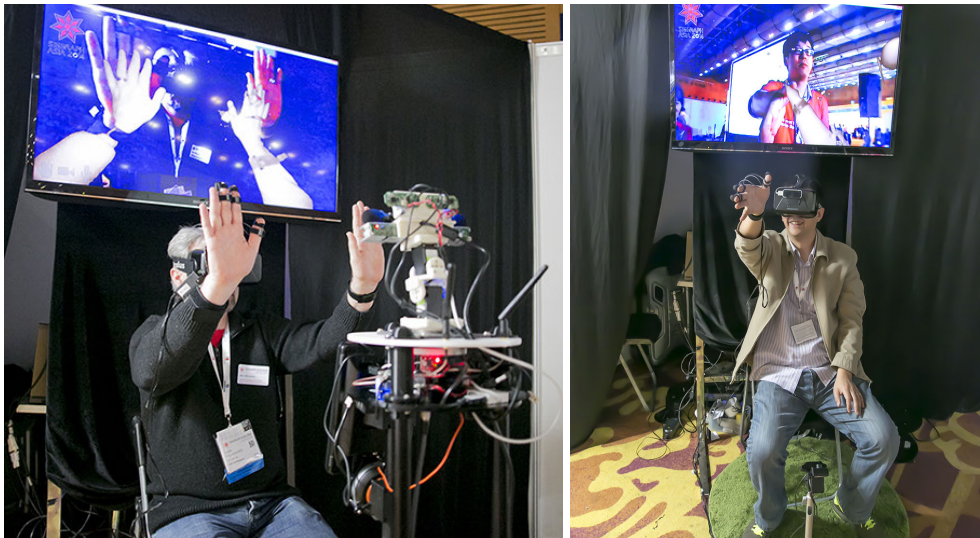


Figure 4.7: Users experiencing Out Of Body and trying to touch their own body from out side (SIGGRAPH Asia 2014).

In an international academic conference (SIGGRAPH Asia 2014) the full system was presented to wide variety of people. Participants were capable to experience their body visuals presented remotely. Also, participants were capable to move closely to their physical body, and experience an Out of Body Experience (OBE). Many reported they found the visuals and interaction very compelling, and OBE visual feedback very interesting. Figure 4.7 shows attendees trying to touch their own body from outside.

In another public event conducted by KMD (KMD Forum), wide range



Figure 4.8: People with different ages testing the system (KMD Forum 2014).

of people had the chance to try the system and experienced their bodies. Even 12 year-old kids were capable to control the motion of the robot. The attendees experienced telepresence in a crowded atmosphere, and communicate with their friends using it.

# Chapter 5

## Reserch Contribution, Limitations, and Conclusion

### 5.1 Research Contributions

The presented work mainly contribute in Telexistence based environments, it studies the important of body presence in a different place, it explored the previous works and literatures about presence. And it proposes a new concept for delivering body visual and haptic feedback to operating user to realize the idea of body presence. This work is considered as a part of progression embodiment in Computer Mediated Communication, with a vision to eliminate the need of physical systems for communication purpose while preserving the perception of presence.

### System Extensions

The system also contributes for other telexistence platforms that does not have any physical arms or hands. The design of the system, and the isolation between the operator system and robot system allowed expanding it to other telexistence platforms. The software design defines an abstraction for robot operation and control<sup>1</sup>, thus by reimplementing the abstract interface for the robot control, its possible to access that robot from the same software the user uses for connecting, and the robot will then sup-

---

<sup>1</sup>See the appendix for UML design of the robot software.

## 5.2. System Limitations

---

port virtual embodiment. This system has been integrated with a previous Telexistence system: TORSO [42]. TORSO robot's main advantage compared with the proposed robot is the 6DOF design of the head, allowing the user to experience parallax motion when moving his head and body. Figure. 5.1 shows the system integrated with TORSO, the user is capable to see his arms when trying to grasp an object on the table. The design



Figure 5.1: TORSO 6DOF Telexistence system. User experience virtual embodiment using TORSO system.

of this system also has the capability to expand to non-vehicle platforms, such as flying systems (drones), or even for diving robots.

## 5.2 System Limitations

The system was designed for telecommunication purposed over telexistence platforms that does not contain any physical limbs, and only has the representation of human's upper body, and head. The absence of physical arms in the robot sides does not allow the user to do any physical manipulation of remote objects. Also because of that, people in the remote place does not have the awareness of the user's arms and hands. A basic solution for this problem was by adding a front screen in the robot side, that displays user's front captured video.

Some technical limitations we faced in the implementation was the 160ms latency of video encoding, decoding and streaming over the network. With internet based environment, this latency may increase depending on the status of the network. To solve this issue, its possible to detect the Quality of Service (QoS) and based on it the video quality is changed to adapt to the network. Realizing this can be using Real-Time Control Protocol (RTCP). However due to the limited time, this was not implemented.

## 5.3 Conclusion

In this research, we presented a novel method to substitute our physical bodies into remote places using Telexistence systems and Virtual Embodiement. We proposed the concept of Virtual Embodiement as a digital representation of objects in a physiscal world, and we focused on the human hands and fingers. The result is a significant reduction in size of telexistence platforms enabling them to be actively used in remote social communication. In this thesis, we discussed the design of Virtual Embodied Telexistence System, and presented a complete system implementation with a framework to expand its functionality into other Telexistence platforms, and we evaluated the proposed method in a lab environment as well as a subjective evaluation in public.

Virtual Embodiement concept means to replicate the Visual, Functional, and Behavioral attributes of a physical object into a digital representation, and to be accessed over cyberspace. Combining Virutal Embodiement with telecommunication tools means the ability to access the digital version of the physical objects over cyberspace. Adapting this concept to human oriented functions creates wide range of possibilities in alternative realities and telecommunications. As humans, we believe our presence in a certain place after we confirm the existence of our bodies, and to confirm its existence we first look at our body and validate it through the visual feedback.



We understand the presence when we navigate and trying to touch the objects within it. Those validations are the key elements for realizing Human Virtual Embodiement.

In this work, we realized the concept of Virtual Embodiement by implementing the visual and haptic feedback. Two methods for body visuals were discussed, one using a computer generated version of the hands, and the other by using image-based capturing and presentation of the hands. For haptic feedback, we introduced a novel method for finger touch estimation using depth array sensor only. Virtual Embodiement in Telexistence helps to reduce the physical requirements to build a telexistence system with visual and haptic feedback of human body.

# Bibliography

- [1] A. Romiszowski and R. Mason, “Computer-mediated communication,” in *Handbook of research for educational communications and technology 2*, 1996, pp. 397–431.
- [2] J. Short, E. Williams, and B. Christie, “The social psychology of telecommunications,” 1976.
- [3] S. G. Jones, “Understanding community in the information age,” *Cybersociety: Computer-mediated communication and community*, vol. 10, no. 35, pp. 219–40, 1995.
- [4] M. Minsky, “Telepresence,” 1980.
- [5] S. Tachi, *Telexistence*. World Scientific Publishing Co., Inc., 2009.
- [6] R. J. Jacob, A. Girouard, L. M. Hirshfield, M. S. Horn, O. Shaer, E. T. Solovey, and J. Zigelbaum, “Reality-based interaction: a framework for post-wimp interfaces,” in *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 2008, pp. 201–210.
- [7] N. A. Streitz, P. Tandler, C. Müller-Tomfelde, and S. Konomi, “Roomware: Towards the next generation of human-computer: Interaction based on an integrated design of real and virtual worlds,” *Human-Computer Interaction in the New Millenium*, Addison Wesley, pp. 551–576, 2001.
- [8] F. Biocca, “The cyborg’s dilemma: Progressive embodiment in virtual

## Bibliography

---

- environments,” *Journal of Computer-Mediated Communication*, vol. 3, no. 2, pp. 0–0, 1997.
- [9] A. Synnott, *The Body Social: Symbolism, Self, and Society*. Routledge, 1993.
- [10] N. I. Durlach, A. S. Mavor *et al.*, *Virtual reality: scientific and technological challenges*. National Academies Press, 1994.
- [11] F. Biocca, “Intelligence augmentation: The vision inside virtual reality.” 1996.
- [12] J. Benthall and T. Polhemus, *The body as a medium of expression*. Allen Lane, 1975.
- [13] O. Sacks, “The man who mistook his wife for a hat,” 1995.
- [14] N. P. Holmes and C. Spence, “The body schema and multisensory representation (s) of peripersonal space,” *Cognitive processing*, vol. 5, no. 2, pp. 94–105, 2004.
- [15] V. C. Anderson-Barnes, C. McAuliffe, K. M. Swanberg, and J. W. Tsao, “Phantom limb pain—a phenomenon of proprioceptive memory?” *Medical hypotheses*, vol. 73, no. 4, pp. 555–558, 2009.
- [16] M. Slater, M. Usoh, and A. Steed, “Depth of presence in virtual environments,” *Presence*, vol. 3, no. 2, pp. 130–144, 1994.
- [17] T. Schubert and F. Friedmann, “Embodied presence in virtual environments,” in *Visual representations and interpretations*. Springer, 1999, pp. 269–278.
- [18] B. G. Witmer and M. J. Singer, “Measuring presence in virtual environments: A presence questionnaire,” *Presence: Teleoperators and virtual environments*, vol. 7, no. 3, pp. 225–240, 1998.

## Bibliography

---

- [19] M. Lombard and T. Ditton, “At the heart of it all: The concept of presence,” *Journal of Computer-Mediated Communication*, vol. 3, no. 2, pp. 0–0, 1997.
- [20] M. Botvinick, J. Cohen *et al.*, “Rubber hands’ feel’touch that eyes see,” *Nature*, vol. 391, no. 6669, pp. 756–756, 1998.
- [21] D. Dunning, “What is the word on self-motives and social perception: Introduction to the special issue,” *Motivation and Emotion*, vol. 25, no. 1, pp. 1–6, 2001.
- [22] C. Cruz-Neira, D. J. Sandin, T. A. DeFanti, R. V. Kenyon, and J. C. Hart, “The cave: audio visual experience automatic virtual environment,” *Communications of the ACM*, vol. 35, no. 6, pp. 64–72, 1992.
- [23] S. J. Gibbs, C. Arapis, and C. J. Breiteneder, “Teleport–towards immersive copresence,” *Multimedia Systems*, vol. 7, no. 3, pp. 214–221, 1999.
- [24] W. Robinett and R. Holloway, “Implementation of flying, scaling and grabbing in virtual worlds,” in *Proceedings of the 1992 symposium on Interactive 3D graphics*. ACM, 1992, pp. 189–192.
- [25] K. Vaananen and K. Bohm, “Gesture driven interaction as a human factor in virtual environments-an approach with neural networks,” *Virtual reality systems*, pp. 93–106, 1993.
- [26] R. P. Darken, W. R. Cockayne, and D. Carmein, “The omni-directional treadmill: a locomotion device for virtual worlds,” in *Proceedings of the 10th annual ACM symposium on User interface software and technology*. ACM, 1997, pp. 213–221.
- [27] H. Iwata and Y. Yoshida, “Path reproduction tests using a torus treadmill,” *Presence: Teleoperators and Virtual Environments*, vol. 8, no. 6, pp. 587–597, 1999.

- [28] J. J. LaViola Jr, D. A. Feliz, D. F. Keefe, and R. C. Zeleznik, “Hands-free multi-scale navigation in virtual environments,” in *Proceedings of the 2001 symposium on Interactive 3D graphics*. ACM, 2001, pp. 9–15.
- [29] B. Peterson, M. Wells, T. A. Furness, and E. Hunt, “The effects of the interface on navigation in virtual environments,” in *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, vol. 42, no. 21. SAGE Publications, 1998, pp. 1496–1500.
- [30] F. Biocca, “The cyborg’s dilemma: Embodiment in virtual environments,” in *Cognitive Technology, International Conference on*. IEEE Computer Society, 1997, pp. 12–12.
- [31] C. L. Fernando, M. Furukawa, T. Kurogi, S. Kamuro, K. Sato, K. Minamizawa, and S. Tachi, “Design of telesar v for transferring bodily consciousness in telexistence,” in *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*. IEEE, 2012, pp. 5112–5118.
- [32] M. Y. Saraiji, C. L. Fernando, M. Furukawa, K. Minarnizawa, and S. Tachi, “Virtual telesar-designing and implementation of a modular based immersive virtual telexistence platform,” in *IEEE/SICE International Symposium on System Integration (SII)*, 2012, pp. 595–598.
- [33] M. Furukawa, K. Sato, K. Minarnizawa, and S. Tachi, “Study on telexistence lxvi: Tele-operation using eye and hand of telesar v,” in *Entertainment Computing Convergence (in Japanese), EC’11*, 2011, pp. 1–4.
- [34] M. Y. Saraiji, C. L. Fernando, M. Furukawa, K. Minarnizawa, and S. Tachi, “Real-time egocentric superimposition of operator’s own body on telexistence avatar in virtual environment.” in *ICAT*, 2013, pp. 35–39.

- [35] N. A. Dodgson, "Variation and extrema of human interpupillary distance," in *Proceedings of SPIE*, vol. 5291, 2004, pp. 36–46.
- [36] OpenCV. Camera calibration. [Online]. Available: <http://docs.opencv.org/>
- [37] I. Poupyrev, M. Billinghurst, S. Weghorst, and T. Ichikawa, "The go-go interaction technique: non-linear mapping for direct manipulation in vr," in *Proceedings of the 9th annual ACM symposium on User interface software and technology*. ACM, 1996, pp. 79–80.
- [38] Y. Yokokohji, R. L. Hollis, and T. Kanade, "Wysiwyf display: A visual/haptic interface to virtual environment," *Presence: Teleoperators and Virtual Environments*, vol. 8, no. 4, pp. 412–434, 1999.
- [39] R. A. Newcombe, A. J. Davison, S. Izadi, P. Kohli, O. Hilliges, J. Shotton, D. Molyneaux, S. Hodges, D. Kim, and A. Fitzgibbon, "Kinect-fusion: Real-time dense surface mapping and tracking," in *Mixed and augmented reality (ISMAR), 2011 10th IEEE international symposium on*. IEEE, 2011, pp. 127–136.
- [40] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Computer Vision, 1998. Sixth International Conference on*. IEEE, 1998, pp. 839–846.
- [41] K. Minamizawa, S. Fukamachi, H. Kajimoto, N. Kawakami, and S. Tachi, "Gravity grabber: wearable haptic display to present virtual mass sensation," in *ACM SIGGRAPH 2007 emerging technologies*. ACM, 2007, p. 8.
- [42] K. Watanabe, I. Kawabuchi, N. Kawakami, T. Maeda, and S. Tachi, "Torso: Development of a telexistence visual system using a 6-dof robot head," *Advanced Robotics*, vol. 22, no. 10, pp. 1053–1073, 2008.

# Appendix A

## Software Implementations

### A.1 Class Diagrams

Here is a list of the main classes used by the user and robot sides. The diagram format used is UML<sup>1</sup>. The code implementation was done using C++ programming language due to efficient performance, and support most of operating systems (There will be no code listing here).

#### A.1.1 User Class Diagrams

Operator software can be abstracted into the main blocks that handles the different types of controllers, and the virtually embodied hands blocks. Figure A.1 shows the relations between the classes.

#### A.1.2 Robot Class Diagrams

The main two blocks in the robot software design are:

- The communication handlers (Figure. A.2) which send/receives the control streams to/from the user side and handles the communication with the robot.
- The robot controllers (Figure. A.3) which creates the abstraction between the operating software, and the running robot. Using this abstraction, the support to different types of robots is possible.

---

<sup>1</sup>Unified Modeling Language: Commonly used by Software Engineers.

## A.1. Class Diagrams

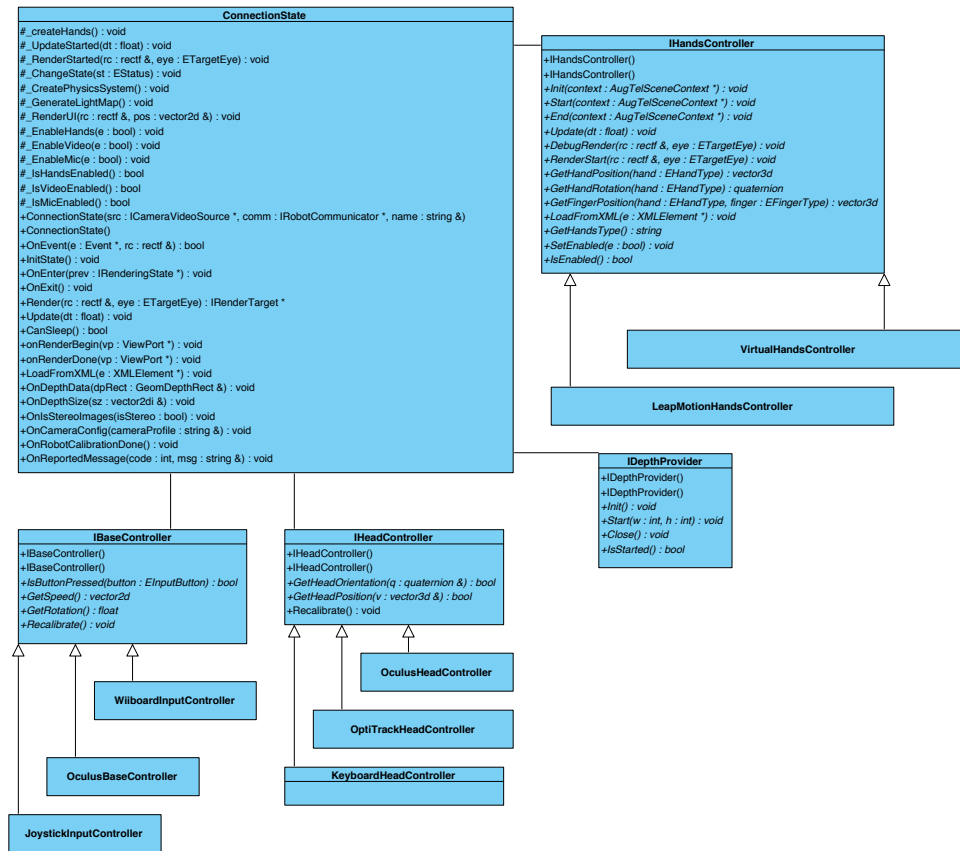


Figure A.1: User side main components class diagram



## A.1. Class Diagrams



Figure A.2: Robot communication handler class diagram

## A.1. Class Diagrams

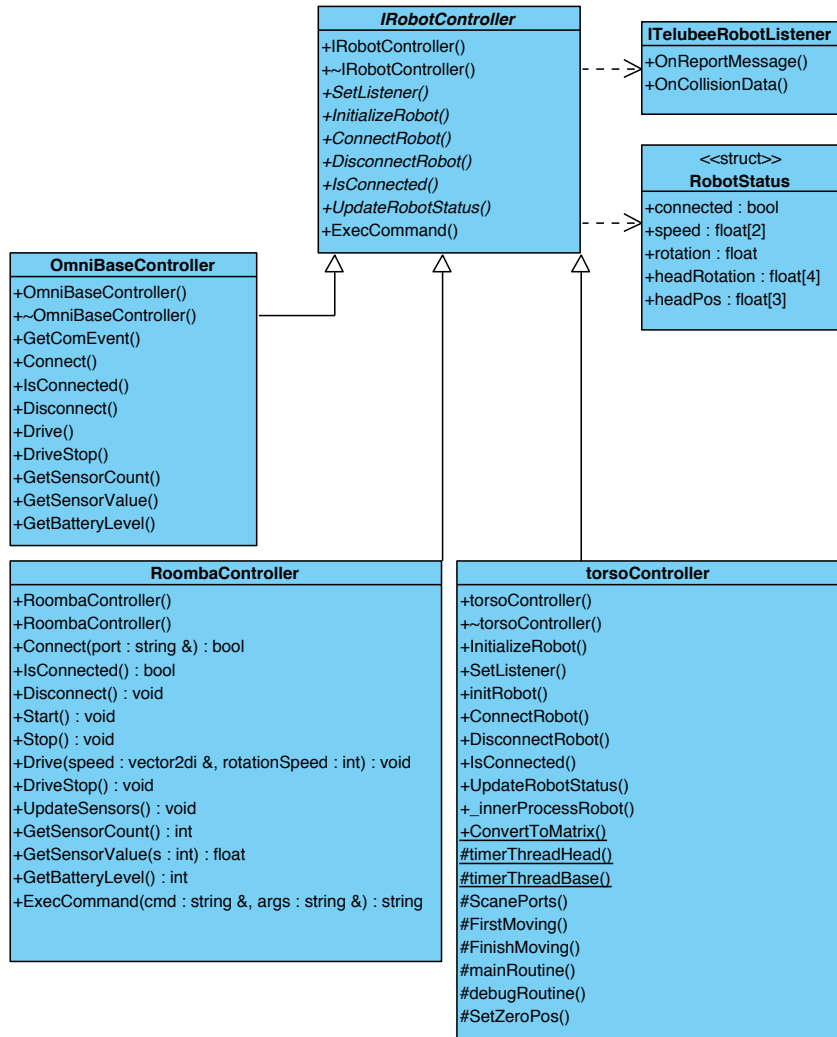


Figure A.3: Robot control class diagram

## A.2 Code Listings

### A.2.1 Camera Lens Correction

The camera lens correction process was written using HLSL language and it runs completely on the GPU. This results very high speed processing, and avoid using any CPU processing power. In the listed code, LensParams1 and LensParams2 represents the calibrated parameters for the lens. Those parameters are lens/camera dependant.

```
struct VS_OUTPUT {
    float4 Pos: POSITION;
    float2 texCoord: TEXCOORD0;
};
VS_OUTPUT main_vp(float4 Pos: POSITION) {
    VS_OUTPUT Out;
    Out.Pos.xy=2*sign(Pos.xy)-1;
    Out.Pos.z = 1.0;
    Out.Pos.w = 1.0;
    Out.texCoord.xy = 0.5 * (1 +Out.Pos.xy);
    return Out;
}
half4 main_fp(float2 texCoord: TEXCOORD0,
    uniform sampler2D image, // input image from camera
    uniform float2 lensCenter,
    uniform float2 Scale,
    uniform float2 ScaleIn,
    uniform float4 LensParams1,
    uniform float4 LensParams2
) : COLOR {
    float2 centerVec=(texCoord-lensCenter)*ScaleIn;
    float rSq=dot(centerVec,centerVec);
    float2 rtheta;
    rtheta.x=centerVec.x/(LensParams1.x+LensParams1.y*rSq+LensParams1.z*rSq*rSq);
    rtheta.y=centerVec.y/(LensParams2.x+LensParams2.y*rSq+LensParams2.z*rSq*rSq);
    float2 tc =lensCenter+Scale*rtheta;
    if(any(clamp(tc,float2(0,0),float2(1,1))-tc))
    return half4(0,0,0,1);
    return tex2D(image,tc);
}
```

## A.3 GStreamer Pipelines

Here is listed the media pipelines that been used to deliver the real-time video/audio contents between the User and the Robot. For encoding/decoding the contents, GStreamer library was used and integrated with the application layer.

The UML Class Diagram for the streamers pipelines is shown in Figure A.4 and for the players pipelines are shown in figure A.5.

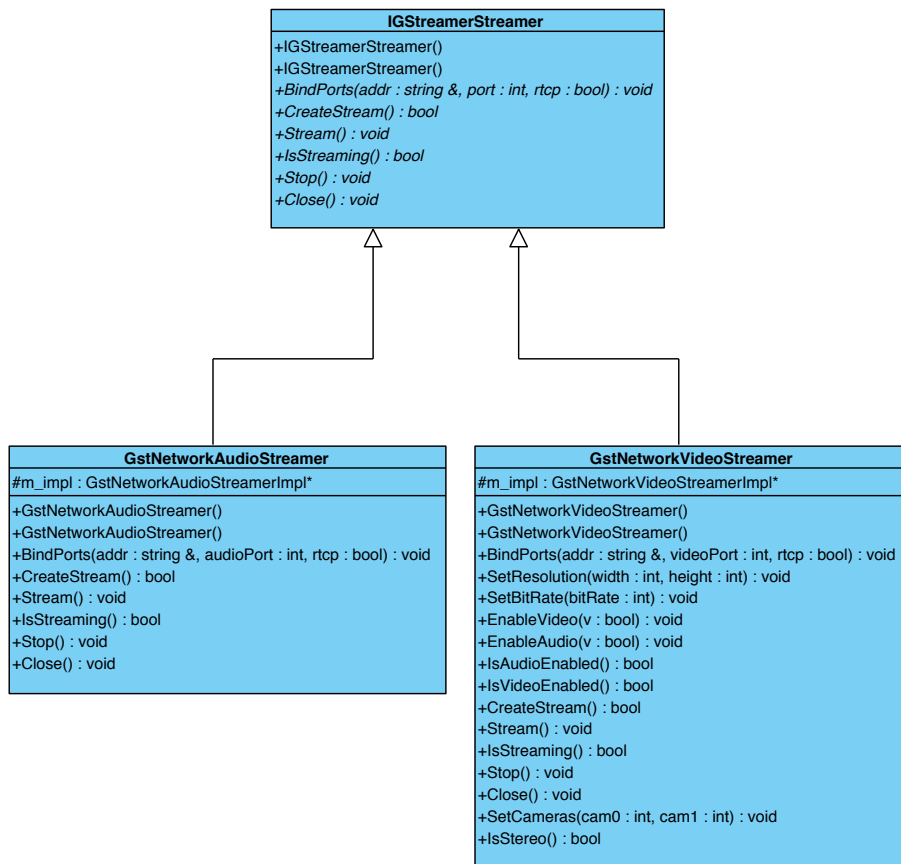


Figure A.4: Video/Audio Streamers Class Diagram

### A.3. GStreamer Pipelines

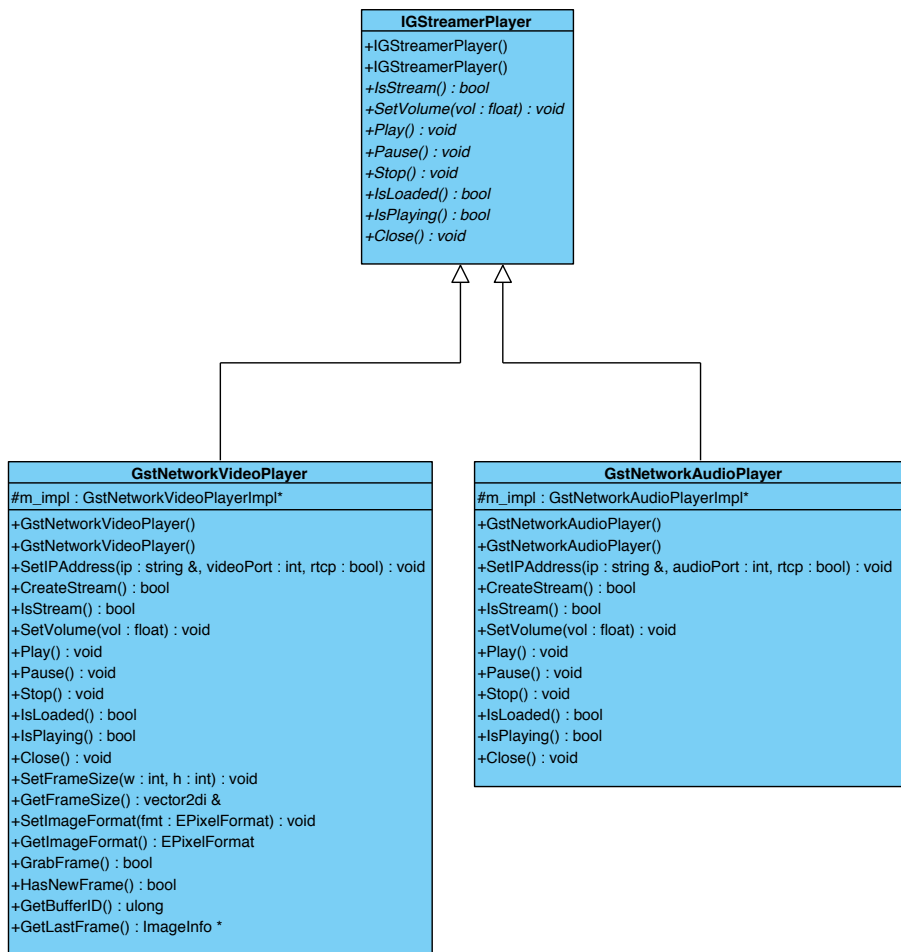


Figure A.5: Video/Audio Players Class Diagram

### A.3.1 Video Pipelines

For video streaming and playback, H264 encoder (or MPEG-4 AVC) has been used due to its efficient performance and reliability in networked environments. The packets of the stream were payloaded via an rtp encapsulation to ensure data delivery over udp.

#### Streaming Pipeline

```
int targetWidth=1920;
int targetHeight=1080;
string cameraFormat=video/x-raw, format=I420, width=targetWidth,
height=targetHeight, framerate=30/1;
string resizeMode=video/x-raw,format=I420, width=targetWidth/2,
height=targetHeight;
videomixer name=mix sink0::xpos=0 sink0::ypos=0 sink0::alpha=1
sink0::zorder=0 sink1::xpos=0 sink1::ypos=0 sink1::zorder=1
sink2::xpos=targetWidth/2 sink2::ypos=0 sink2::zorder=1
//fill background with black
videotestsrc pattern="black" !
video/x-raw,format=I420,width=targetWidth,height=targetHeight
! mix.sink0
// first video source (web camera)
ksvideosrc name=src1 device-index=0 ! cameraFormat!
videoconvert ! videoscale ! resizeMode ! mix.sink1
//second video source (webcamera)
ksvideosrc name=src2 device-index=1 ! cameraFormat !
videoconvert ! videoscale ! resizeMode ! mix.sink2
// Send the composed videos to encoding and streaming
! x264enc name=videoEnc bitrate="SelectedBitrate"
speed-preset=superfast tune=zerolatency sync-lookahead=0
pass=qual ! rtph264pay ! udpsink name=audioSink host="TargetHost"
```

#### Receiving Pipeline

```
udpsrc caps=application/x-rtp,media=(string)video,
clock-rate=90000, encoding-name=H264 ! rtph264depay
! avdec_h264 ! videoconvert ! video/x-raw, format=RGB
! appsink
```

### A.3.2 Audio Pipeline

For the current implementation, AMR <sup>2</sup> was used due to its efficient compression for speech coding.

#### Streaming Pipeline

```
directsoundsrc! audio/x-raw,endianness=1234, signed=true, width=16,  
depth=16, rate=8000, channels=1 ! amrnbenc ! rtpamrpay  
! udpsink name=audioSink host="TargetHost"
```

#### Receiving Pipeline

```
udpsrc name=audioSrc  
caps=application/x-rtp,media=audio,clock-rate=(int)8000,  
encoding-name=AMR,encoding-params=1,octet-align=1  
! rtpamrdepay ! amrnbdec ! directsoundsink
```

---

<sup>2</sup>Adaptive Multi-Rate audio codec