MeBot: A robotic platform for socially embodied telepresence

by

Sigurður Örn Aðalgeirsson

BSc. Electrical and Computer Engineering, University of Iceland (2007)

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning,
in partial fulfillment of the requirements for the degree of

Master of Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September 2009

Author ______________________________
Program in Media Arts and Sciences
August 7, 2009

Certified by ______________________________
Dr. Cynthia Breazeal
Associate Professor of Media Arts and Sciences
Program in Media Arts and Sciences
Thesis Supervisor

Accepted by ______________________________
Dr. Deb Roy
Chair, Departmental Committee on Graduate Students
Program in Media Arts and Sciences

MeBot: A robotic platform for socially embodied telepresence

by

Sigurður Örn Aðalgeirsson

Submitted to the Program in Media Arts and Sciences,
School of Architecture and Planning,
on August 7, 2009, in partial fulfillment of the
requirements for the degree of
Master of Science

Abstract

Telepresence refers to a set of technologies that allow users to feel present at a distant location, telerobotics is a subfield of telepresence. Much work has been done in telerobotics through the years to provide safer working environments for people or to reach locations that would otherwise be inaccessible. More recently telerobots have been developed for communication purposes but as of yet they have not accommodated for other channels of communication than audio or video. The design and evaluation of a telepresence robot that allows for social expression is the content of this thesis. Our claim is that a telerobot that communicates more than simply audio or video but also expressive gestures, body language or pose and proxemics will allow for a more engaging and enjoyable interaction. An iterative design process of the MeBot platform is described in detail as well as the design of supporting systems and various control interfaces. A human subject study was conducted where the effects of expressivity were measured. Our results show that a socially expressive robot was found to be more engaging and likable than a static one. It was also found that expressiveness contributes to more psychological involvement and better cooperation.

Thesis Supervisor: Dr. Cynthia Breazeal

Title: Associate Professor of Media Arts and Sciences, Program in Media Arts and Sciences

MeBot: A robotic platform for socially embodied telepresence

by

Sigurður Örn Aðalgeirsson

The following people served as readers for this thesis:

Thesis Reader____

Dr. David Small
Associate Professor of Media Arts and Sciences
Program in Media Arts and Sciences

Thesis Reader____

Dr. Joseph A. Paradiso
Associate Professor of Media Arts and Sciences
Program in Media Arts and Sciences

Acknowledgments

First of all, I would like to thank my advisor Cynthia Breazeal for taking a chance on me and accepting me into the amazing group of students and staff that is the Personal Robots group at the Media Lab. Her critical eye and creative ideas made my work stronger. I would also like to thank my readers David Small and Joseph Paradiso whose work has been inspirational to me for a long time, your input on this thesis was valuable

I would like to thank the Media Lab for sponsoring me to do my research as well as providing me with an excellent opportunity and environment to learn. Nokia deserves thanks for their generosity at the initial stages of the project.

The Personal Robots group is packed with creative talent and I thank all of them for creating a very interesting and fun work environment. Special thanks go to Dan Stiehl, Matt Berlin, Jesse Gray, Mikey Siegel, Ryan Wistort and Guy Hoffman for their critical input on this thesis and general companionship.

I enjoyed working with many very talented undergraduate researchers in the past two years. Many of them contributed excellent work towards this thesis. Special thanks go to Yingdan Gu, Amy Qian and Kristopher Dos Santos for their help with the mechanical design, Daniel Soltero for his help on electrical systems and Nancy Foen for great electrical design and invaluable input on the design and execution of the study.

I thank all of my friends, old and new, for their support throughout my studies. I especially thank my girlfriend Carnaven for her support and for putting up with me for these past months of intensive work.

My loving family has supported me every step of the way. Love to my grand parents, parents and siblings. Alli pabbi, mamma, Villi pabbi, Elísabet, Viktor, Sigrún, Aníta, Íris, Gísli og Þórey, ég elska ykkur öll.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

The aim of telepresence research is fundamentally to allow people to be in two places at the same time. There are many reasons why we might want to occupy two spaces at the same time, examples include wanting to provide safer working environments, perform surveillance, attend meetings or simply spend time with our loved ones. Different situations pose different requirements of the communication medium and therefore many different telepresence systems with different capabilities have been developed.

Face to face is still the golden-standard in communication, against which all platforms are compared. This is partly due to the rich set of social behaviors and cues that we as humans know and share. These behaviors can be viewed as compressed packages of information. The whole package need not be sent between two people that share the same repertoire of social behaviors, rather only a reference to it in form of a pose, gesture, movement, intonation of voice or by other means.

The channel for these social cues is severely limited when people communicate via telephones or videoconferencing, possibly resulting in less efficient or less enjoyable interactions. The reason why social cues are hard to convey using these media is because they are composed of a complex combination of motion, body language and expression as well as spoken language. These properties are hard to capture at the source but even harder to reproduce at the destination using existing systems.

Beyond the discussion of communication efficiency, presence plays a very strong part in all of our interactions with other people. Some individuals seem to have a very strong presence, meaning that they rarely go unnoticed when they enter a room and when they speak people tend to listen. Others don't seem to posses that ability. In communications, and especially in remote collaborations, presence plays a very important role in establishing hierarchy. A person that takes part in a meeting using phone-conferencing or even video-conferencing is not very well represented in the remote space and thus might get less attention or not be able to make as big of a contribution as if they were there in person. Providing a remote participant with adequate embodiment has the possibility to increase their level of engagement in the meeting and contribution to it as well as possibly making the experience more enjoyable.

Telepresence robotics for communication purposes is an idea that has been around for a while now but to the author's knowledge, no serious effort has been put into developing socially expressive systems. *Embodiment* and *immersion* are concepts that are used to describe the level of presence that people interacting with the robot experience and the level of engagement or involvement the operator experiences respectively. Many systems focus on providing deep levels of immersion and much research has gone into haptic feedback systems and the like towards that goal. Embodiment has been the focus of many systems for different purposes, some applications require a high level of dexterity at the remote site and therefore systems are developed that provide high resolution in motion. Telerobots meant for communication need to embody the operator in a way that provides them with adequate representation in the remote space. It is the belief of the author that a socially expressive embodiment is needed.

1.2 Thesis Statement

It is my claim is that affording telepresence systems the ability to convey their operators' non-verbal behavior, gestures, body language and proxemics, can make remote interactions more efficient, more engaging and more enjoyable.

The design of a system that is capable of social expression is presented as well as an evaluation of the validity of this claim.

1.3 Thesis Overview

The first chapter of this thesis gives the motivations for the work as well as pose the thesis question. This chapter also describes previous work in the field of telepresence and other related fields.

In the second chapter, we go into detailed descriptions of the design process for the robot. Every major prototype is described in terms of its mechanical design, electronic design and software design. Lessons learned from every prototype are discussed. The design of a motor control scheme *MCB Mini* is described in detail, as are some fundamental concepts as they apply to motor control. The interfaces for controlling the robot are described as well as supporting systems that were developed.

The third chapter describes the evaluation of the work. The design and execution of an experiment for evaluating the MeBot is explained as well as a discussion provided of which metrics were thought to be important to observe. The results of the study are presented and a discussion of those results.

In the fourth chapter we give an overview of the presented work and discuss its impact. The results of the evaluation are further discussed and possible directions for future work are provided.

The appendix contains documents from the evaluation of the system such as the questionnaire and list of items used.

1.4 Previous Work

1.4.1 History Lesson

Research has been conducted in the field of telepresence for a long time. One of the oldest project in the field dates back to 1952 when Goertz designed his mobile manipulator intended for handling hazardous radioactive materials at a safe distance [Goertz, 1952]. The first systems were mechanical mechanisms for manipulation and thus had a very limited reach in the range of a few meters. The motivation for these early systems was to provide safer working environments or to access locations otherwise not available to people. Later several telepresence projects were developed for solving similar problems of manipulation to allow people to work under difficult conditions. Examples include working at great depths in the sea [Yoerger & Slotine, 1987], in microscopic workstations [Codourey *et al.*, 1997] and performing space-station maintenance [Li *et al.*, 1996].

Marvin Minsky published an excellent article in 1980 [Minsky, 1980] where he discussed advances that had been made in the field of robotics, and AI as they applied to telepresence. He discussed possible applications for telepresence robots and accurately predicted their use in many fields such as deep-sea- and space-exploration, nuclear power plant operation and surgical operations. His discussion spanned wide and far, covering an impressive amount of topics and providing a good historical background. The author found it notable that he never mentioned telepresence for the purpose of social communication.

1.4.2 Telerobots for Human Interaction

It was not until the mid-nineties when research in this field started looking at telepresence for the sole purpose of communication. Some of the first experiments were performed by Eric Paulos and John Canny at UC Berkeley [Paulos & Canny, 2001], [Paulos & Canny, 1998]. Their initial telerobots were blimps fitted with webcams, microphones and speakers but later they developed the *Personal Roving Presences* or PRoPs which allowed people to roam about an office and provided some embodiment.

Much research effort has gone into telerobotics for healthcare. Examples include home care assistance [Michaud *et al.*, 2006], interpersonal communication for elderly care [Tsai *et al.*, 2007] and a robotic teddy bear for early education, family communication and therapeutic purposes [Stiehl *et al.*, 2006].

Reserchers performed a study where they compared people's reactions to a person mediated through a normal videoconference application versus a highly expressive and anthropomorphic android. Results show that people experienced much stronger presence with the android but that they also felt it was very uncanny [Sakamoto *et al.*, 2007].

Field trials have been performed to evaluate the public's acceptance of teleoperated service robots in public places like malls or subway stations [Koizumi *et al.*, 2006]. For this particular experiment, researchers developed autonomously generated non-verbal behavior to accompany speech from the operator.

1.4.3 Social Robots

A comprehensive but slightly dated survey of the research in social robotics was presented by Fong et al [Fong *et al.*, 2003]. In their conclusions they discussed the effects of emotion, appearance and personality of robots. These issues are relevant in the design of telerobots although the general notion is that telerobots should convey a person's emotion, appearance and personality as opposed to expressing these qualities on its own. A detailed discussion on the design of social robots, why social behaviors should be incorporated into robotics, challenges and proposed solutions are presented in [Breazeal, 2004]. A discussion of the different classes of social robots can be found in [Breazeal, 2003]. An effort was made to develop a conversational robot (autonomous system, not for communication) that made use of non-verbal channels like facial expression, pointing and posture [Tojo *et al.*, 2000]. The researchers showed that people who had conversations with the robot when it was expressive reported a higher level of conversational turn-taking, more natural behavior and more natural utterances than people who conversed with a static robot.

1.4.4 Specific Telepresence Research Topics

Notion of Presence

In the domain of embodied agents, researchers have looked at designing embodied, socially aware, intermediary agents to make cellular communications less intrusive [Marti & Schmandt, 2005]. Researchers have also studied how social codes change in places shared by people physically present and embodied by robotic avatars [Karahalios & Dobson, 2005]. Some early work looked at how adding 3D movement and embodiment that reached out of the computer screen compared with conventional 3D rendered on a 2D substrate [Negroponte, 1983].

Immersive Virtual Environments have been used extensively by researchers in the fields of *Communication* and *Social psychology* to measure different levels of presence and what affects it. Bailenson et al. investigated how virtual reality avatar realism both in behavior and appearance, affected levels of presence-metrics in subjects [Bailenson *et al.*, 2005]. Researchers have shown an increase in the measure of *social presence* between 2D and 3D videoconferencing environments [Hauber *et al.*, 2005]. Increase in *social presence* and *interpersonal trust* has been shown to result from the use of virtual avatars for net-based collaborations [Bente *et al.*, 2004].

Inherent Time Delay

Time delay is an unavoidable aspect of any communications system. Delay between action and observed effect is a known issue in the *Control Systems* literature and can cause feedback systems to become unstable. When a human teleoperator is controlling a robot at a distance, they are essentially a part of a feedback loop and careful consideration of the effects of delays needs to be taken. An interesting investigation into the effects of time-delay in teleoperated systems can be seen in [Lane & Akin, 2001] where predictive displays were used to minimize the effect of delays. Other researchers have developed state-space models of telemanipulators and delay models of communication channels and used the models to predict the manipulators reactions to control command, resulting in a tighter human control loop [Brady, 2001]. The concept of *adaptation* as it applies to dealing with observed time delay between action and effect is introduced in [Held & Durlach, 1991].

Interfaces

Controlling robots that are located far away is a difficult task and researching methods to do so effectively is a worthy effort. Glas et al. focused on developing a method for teleoperators to control multiple robots at the same time. The robots were intended for giving directions in a shopping mall and they operated semi-autonomously. The system basically defined a priority of situational severity and prompted an operator to "help the robot out" whenever a dire situation was detected [Glas & Ishiguro, 2008].

Much work has focused on providing an immersive experience for the operator in the hope that the feeling of really "being there" will elicit better control. A good overview of haptics for telepresence is provided in [Fisch *et al.*, 2003]. Research into multi-camera fusion and fusion of multiple sensory modalities for more immersive control was conducted in [Keyes *et al.*, 2006] and [Keyes, 2007].

Novel interface techniques were evaluated for a telerobot intended for home care assistance [Michaud *et al.*, 2006]. Virtual reality augmented displays with embedded real-time video for increasing the operator's FOV were developed as well as different navigational command strategies evaluated.

1.4.5 Overview of Commercially Available Systems

Quite a few telepresence robots have been developed for commercial purposes. These systems are usually marketed for remote meetings, home security or entertainment. The author chooses to group commercially available telerobots by size; this is not only because of their obvious visual differences but also because of their designers' choice of means to embody the operator.

Some argue that an important part of embodiment is human form, particularly human dimensions and size. There are several telerobots that have been built to satisfy these requirements. One of the more advanced systems is the QA robot from *Anybots.* This robot is built on a segwaying base and has close to human size and proportions. It has a particularly curved and visually appealing appearance. The robot uses a LIDAR for navigation, has cameras mounted in a movable head and an embedded display in the "chest". Other human-scale telerobots include the *Giraffe* and the *RP-7* (*Dr. Robot*). The *Giraffe* is a product of *HeadThere* and is mostly designed for remote meeting situations, it basically provides a mobile base with no sensing and two DOFs for controlling height and head tilt. *Dr. Robot* is designed by *InTouch Health* and its main application is allowing doctors to visit patients in other hospitals. It has FDA approval to interface with certain classes of medical devices and can for example interface with some medical sensing equipment and allow the doctor to monitor a patient's signals while remotely examining them.

Larger platforms can sometimes suffer from any or some of the following problems:

- They are often expensive and therefore not applicable to massive deployment
- They usually have a fairly high center of mass and can therefore become unstable and need to move slowly
- They can require a lot of power and therefore have short battery life
- Human form raises expectations for human ability and in many cases the robots fail to live up to them (have difficulty passing door sills etc.)

This has pushed many companies to focus on developing smaller platforms that are portable and designed for table top usage. Notable systems include *iRobot's ConnectR*, *WoWee's Rovio* and *Spykee* from *Meccano*. The *ConnectR* is basically a modified version of *iRobot's* famous vacuum cleaner, the *Roomba*, but with a camera, microphone and speaker. The *Rovio* has three omni-directional wheels for mobility and is fitted with a microphone, webcam and a speaker. Its most impressive technology is its navigational ability but it uses a neat IR marker system for finding its way. *Spykee* offers a very cool and sleek look and is mostly marketed for entertainment.

Chapter 2

Design and Implementation

2.1 Initial Motivations

When people communicate face to face, they don't only listen to the words being spoken and interpret them directly, people are constantly evaluating their meaning based on the context of the conversation, tone of the speaker's voice or other signals that can possibly add more information or meaning to the conversation. Much of this information is lost when people communicate through the narrow channel of cellular networks but we have become adjusted to it and simply know what to expect from that particular medium for communication.

The ideas that sparked the MeBot project were those of expressive cellphones. Imagining cellphones that could channel more information than just audio or possibly also video. In addition, they could convey the caller's internal emotional state like excitement or sadness in a subtle way via posture or expressive movement as well as establish the level of privacy of a conversation by adjusting personal proximity amongst other things.

In the beginning of this project we put a couple of raw ideas down for the robot's appearance and how it would function in a world without constraint. These ideas would later become the design guidelines or goals for the project, subject to compromises when it came to prototyping for the purposes of testing and evaluating. The initial ideas for the look and size of the robot were that it should essentially act as an exoskeleton for the cellphone, that is to say that it would be on the same size scale as the phone while providing it with the mechanism for motion. The structure of the robot should be as minimal as possible so that the phone could act as a source of power for the robot. The idea was that the robot wouldn't have any intelligence on its own but rely purely on the computational power of the cellphone. Without the phone, the robot wouldn't be able to do anything or even look like it could do anything, the whole idea was that it would only act to mobilize your mobile phone as well as provide expressive movement. The movement of the robot would be generated either by pre-recorded animations that could be triggered either automatically by sensed events or manually by the operator. The robot's movement could also possibly be directly controlled by mapped sensory input from the operator.

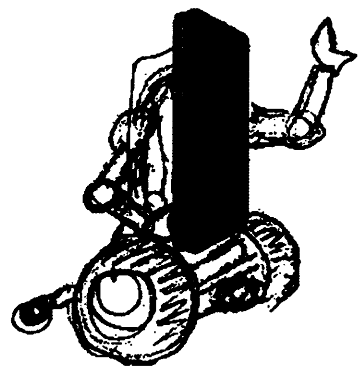

Figure 2-1: A rough concept sketch from one of the first design meetings.

In the original designs there were thoughts of having users generate their own set of automatic behaviors for the robot. Every user could choose the behavioral characteristics of their robotic avatar by either creating idle gestures or signature movements like dances or expressions. When a phone call would be received on a cellphone inserted in a robot, it would be accompanied by a set of the caller's specific behaviors and the robot would then effectively "become" that caller in the sense that it would use the expressions, poses and other behaviors that caller has programmed or recorded. A cellphone, when augmented with this robotic exoskeleton would then become a host that can be occupied by the many different callers in a stronger way than it could before.

2.2 Robot Design

2.2.1 Specifications

As was discussed earlier, we had specific ideas for how the robot should look and function when we started developing this project. Some specifications for the design of the robot were set forth so that later, when we were engaged in the design, we wouldn't lose sight of the original design objectives. Of course as the project advanced, new specifications were conceived and adjustments were made to existing ones as compromises had to be made.

Gesture

A telepresence robot meant for enabling social expression will have to have means to convey expression. Non-verbal expression is mostly thought of as arm or head movements but this only applies if we choose to go with a human body analogy in our design. The robot could be designed to take the shape or form of any creature, possibly even something that has never been seen in nature. This could very well prove to be beneficial to the end goal of the system, as known forms like the human one come loaded with expectations from users which need to be met.

Posture

People's body language alone can tell us a lot about the context of an ongoing conversation, its content and its participants' feelings about it. The posture of a telepresence robot should allow for these messages to be conveyed, that is to say that the design of the robot should facilitate freedom for expression through body language.

Mobility

Mobility will be an essential feature of the telepresence system as it can provide a form of expression that is important in many interactions. Through the adjustment of inter-personal proximity, the operator can set the tone of the conversation as well as send a message to others either to join or to stay away depending of the conversation's level of privacy. Research that is concerned with these behaviors refers to them as *proxemics*.

The operator would also be free to navigate between multiple different conversations with different partners much like people do in real meetings. They could just as well decide not to talk to anybody but focus their attention on the prototype/documents/model being developed in that meeting.

Portability

Many important interactions in collaborative situations take place outside the meeting rooms. This makes it harder for telepresence systems to allow a remote participant to fully play a part in the collaboration as they usually confine them to a conference room. That is why the portability of this type of a telepresence robot is crucial in maximizing the remote operator's participation in meetings. Portability mostly restricts the size and the weight of the robot but it can also have an impact on the design of the interface.

2.2.2 First Prototype

When we built the first prototype of the robot we were looking to make a simple robot that could possibly convey the effect of the expressive cellphones idea as well as giving us a chance to get feedback and learn which direction this project should take.

Mechanical Design

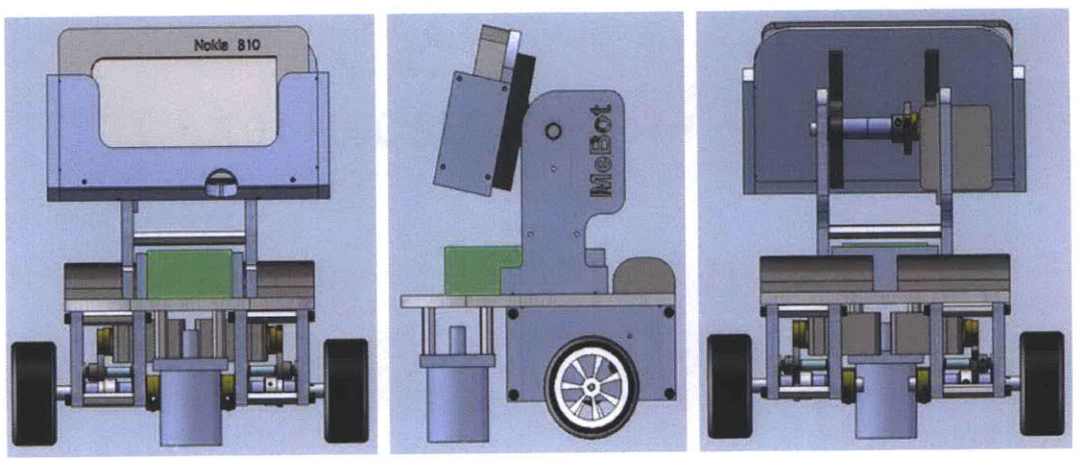

Figure 2-2: A rendering from the mechanical design model. From left to right: Front view, side view, back view.

The mechanical design of the first version of the robot as well as the preliminary designs of the MeBot V2 were made by an undergraduate student under the supervision of the author. The student's name is Yingdan Gu and this project was done as a partial fulfillment of the requirements for her bachelor's degree from the *Mechanical Engineering department* at MIT [Gu, 2008].

The main design goal for this version of the robot was simplicity and compactness. At that point we simply wanted to show that this type of a robot could be portable and small while still delivering the effect of expressiveness. We decided that it was sufficient to provide a two degree of freedom (DOF) mobile base as well as one DOF that would tilt the cellphone up and down. This set of DOFs would allow the user to move around and look at different things as well as displaying some human-like expression such as nodding or shaking of the head.

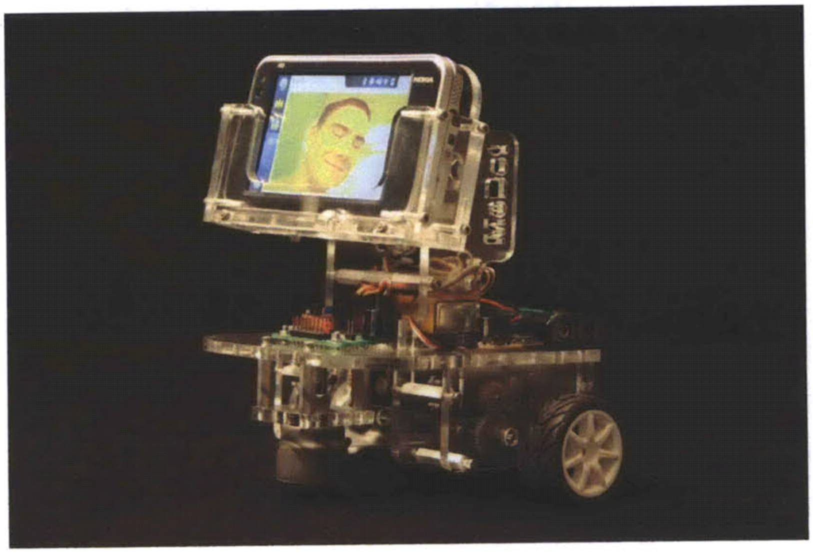

Figure 2-3: A picture of the MeBot V1.

For the sake of simplicity we decided to use servo motors for this version of the robot, the *HiTec* HS-311 model was chosen. The two motors that controlled the wheels had to be modified for continuous rotation. The pan motion of the neck was accomplished by mounting the cellphone cradle to a 0.25" shaft which was directly driven by the servo motor via geared servo-horn and a pinion gear mounted on the shaft. Because of spacing constraints, the wheels could not be directly driven by the servo-horns in the same manner so an intermediary pinion-gear was used to transmit torque to the output shafts.

Software

The device that was chosen for this version of the robot was the *Nokia* N810 which is basically a handheld computer in the form factor of a cellphone. The device runs a customized, lightweight distribution of *Linux* called *Maemo*. At the time, the author was particularly excited about the programming language *Python* and the platform had well supported *Python* libraries to access the device's hardware so we chose to use it for all software in this version.

GStreamer is a library that provides relatively easy manipulation of media streams. It provides a *pipeline* structure that defines *sources*, *filters* and *sinks*. *Sources* can be capture devices, media files or streams arriving at a network port just to name a few examples, *sinks* can be audio or video devices like sound or graphic cards, streams can also be sent into *file-sinks*. *Filters* are modules that can modify streams in many ways, they can for example change the formats of audio streams, compression of video streams etc. Multiple streams can also be multiplexed into one stream, the *multiplexers* and *demultiplexers* are used to handle those types of streams. *GStreamer* has binaries that have been compiled for the *ARM* processor architecture of the N810 device as well as *Python* wrappers which made it a good choice for implementing the audio and video transmission aspect of the project.

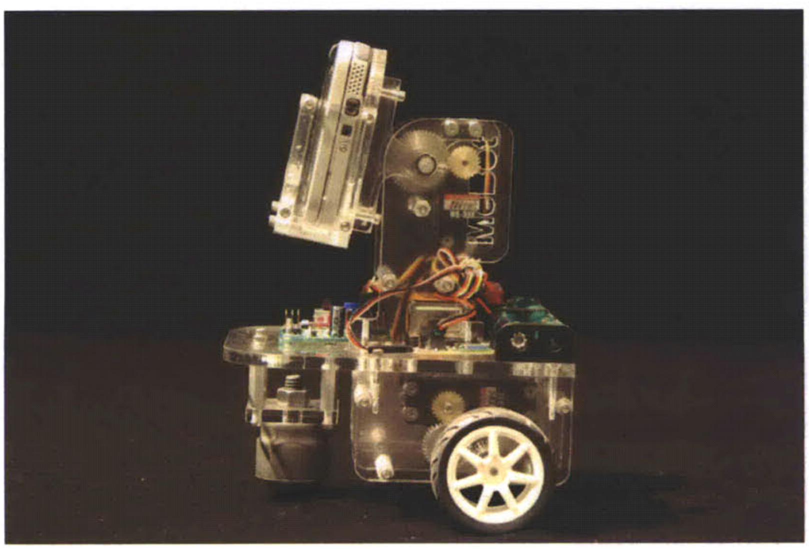

Figure 2-4: A side view picture of the MeBot V1.

To make the connection of two devices easier, a web server was set up to maintain a list of users that had logged in as well as their IP addresses. If a user wanted to make a connection with another user, they would query the server for a list of connected users and select the one they want to connect to. The server would then send the IP address of that user to their device which would establish the connection. Once the connection had been made, the audio and video were streamed using the RTP network protocol but since we could afford packet loss in motor commands, we decided to stream those using the UDP protocol. Once the receiver's device started receiving motor commands, that device started searching for bluetooth devices of a certain name. If the device was found, the motor commands were streamed wirelessly via bluetooth to this device. This bluetooth device was a bt-to-serial bridge that connected the motor control of the robot to the N810 device wirelessly. It will be further explained in the electronics part of this section.

Interface

For this prototype we decided to use N810 devices on both ends of the interaction, that means that the robot was equipped with an N810 and the operator also had one to control the robot. The interface was purely graphical. We decided to make the video stream from the robot occupy the whole screen of the device so to maximize the operators view of the remote scene. The N810 device has a touchscreen and comes with a small pen for navigating menus by clicking on the screen. Allowing the user to control the robot by dragging the pen on the screen that is showing the remote view gave good results and people who tested the robot found it to be surprisingly natural and quite easy to use.

The whole screen was designated for navigational commands but a small strip, a width of about half an inch, on the left side was reserved for controlling the head-tilt. When the pen entered the screen, the x and y positions of the pen were marked, when the pen was lifted off the screen, the distance traveled in x and y was calculated and the y distance was mapped to a certain amount of traversal that the robot should make while the x distance was mapped to an angle that the robot should rotate to. If the drag of the pen happened in the half-inch strip on the left side of the screen then the distance would be mapped to the desired angle of the head-tilt servo.

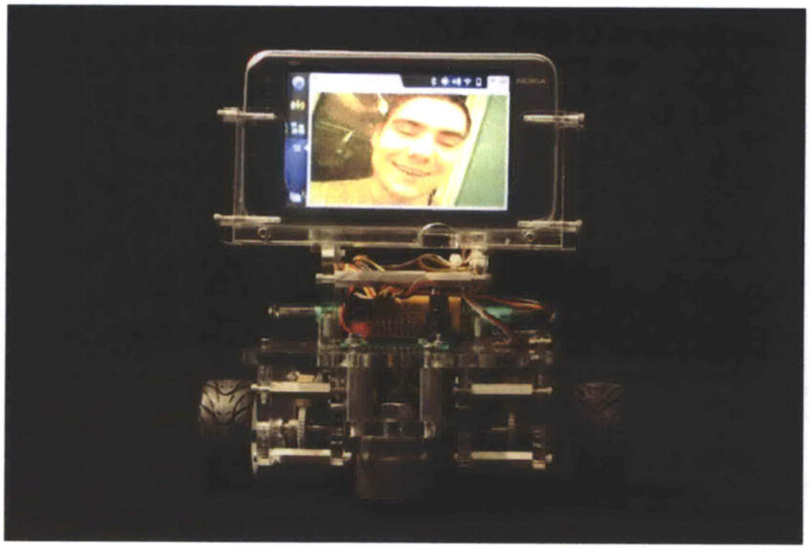

Figure 2-5: A front view picture of the MeBot V1.

Electronics

The electronic design for this robot was simplified by the fact that we could use an off-the-shelf servo motor controller. We used a *Parallax* serial servo controller, this model can control 16 servo motors at once but two boards can be daisy-chained to control up to 32 motors simultaneously. The *Parallax* controller responds to serial commands sent using the RS-232 protocol.

The serial commands were sent from the N810 device through a wireless connection to a custom board that had a bluetooth radio. The radio chip was an RN-41 model from *Roving Networks*, it fully encapsulates the Bluetooth stack and emulates a configurable UART port. The commands were passed down from the N810 device to the bluetooth board wirelessly, they were then passed to the servo controller via serial commands at TTL levels and the controller finally sent target positions down to the actual servo motors via a PWM signals.

Lessons Learned

By building and testing the first prototype of the robot we learned many things about what people liked and disliked regarding the idea of expressive telepresence robots, even though this particular robot wasn't very expressive. It created grounds for discussion and sparked ideas with people that used it. We got a better feeling for what people expected from a robot like this and what it had to be able to do to achieve those goals. A few things that we wanted to change in future versions of the robot are listed below:

- Servo motors produce a lot of noise as well as taking up a lot of space. It became clear to us that our design would benefit from using DC motors with our own sensing and motor control.
- The orientation of the cellphone should be a *portrait* style as opposed to *landscape* because the phone's display was found to symbolize the human head of the operator which has the proportions of a *portrait.*
- Effort needed to be made to increase the expressivity of the robot, arms would be beneficial to achieve this. Also it seemed that people didn't think to use the nod or shake the robot's head so we decided to make head movement more explicit in the next design.

2.2.3 Second Prototype

With the second prototype of the robot we wanted to address some of the issues that we had with the first one as well as incorporating some modifications that we learned about by getting feedback from users.

Mechanical Design

One of the less significant changes we made in this prototype of the robot was switching from acrylic material to delrin. Delrin is a much more durable and stronger material than acrylic. It doesn't crack like acrylic and makes for easier machining and custom modification.

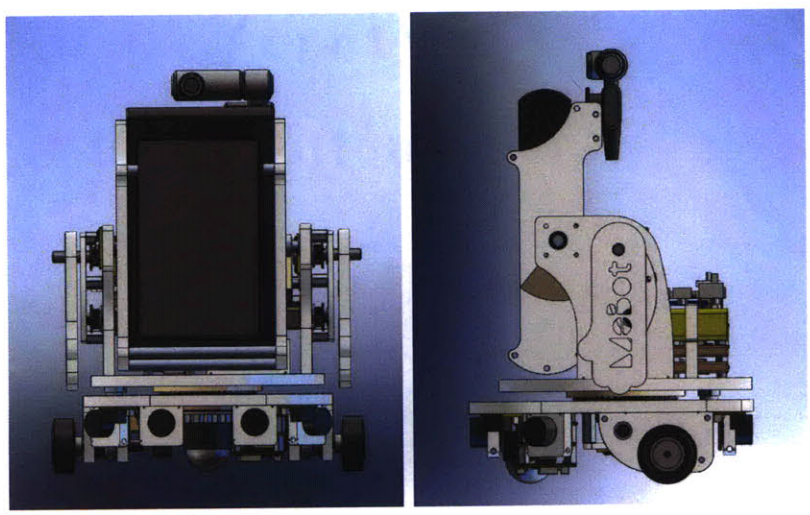

Figure 2-6: A rendering from the mechanical design file for the MeBot V2. Front and side views.

In this version of the robot we used DC motors with custom motor controllers (described in section 2.4.2). This change was probably less visible to the users than many others but one of the more difficult ones and required the most work. We chose to use motors in the GM line from *Solarbotics* for their compactness, easy mounting and small form factor gearboxes.

When the first prototype was designed we thought that a differential drive for mobility would provide the operator with sufficient ability to pan around and increase their field of view. This was found to have a slightly unnatural looking effect, we decided that the robot should have a separate DOF to provide this look-around panning behavior. For simplicity and compactness, this DOF was placed at the base of the robot and the arms and neck were built on top of this rotating platform. The pan mechanism can be viewed in figure 2-6.

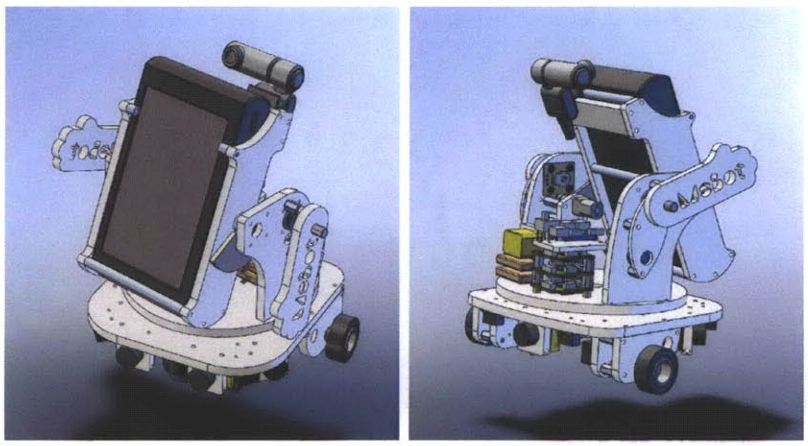

Figure 2-7: A rendering from the mechanical design file for the MeBot V2. Perspective views.

The user feedback from the first version of the robot taught us that people wanted to have some way to reach into the remote space, either to call attention to the robot, direct the attention of others or simply to gesture with speech. We decided to fit this version of the robot with very simplistic arms that could at least begin to address some of those issues. We designed simple 1-DOF arms that allowed the operator to call for attention and gave them some ability to point by combining the arms with the neck pan. The arms were driven by GM *Solarbotics* motors located far down on the shoulder plate because of space issues. Torque was transmitted to the arms through a belt system, the belt system provided quiet transmission as well as acting as a clutch that would simply skip under high load instead of breaking the hardware.

The arms can be seen in figures 2-7 and 2-8, the belt transmission can be viewed to the left in figure 2-6 and in figure 2-8

Device

When designing the software for the first prototype of the robot, it became clear that the N810 did not have the computational resources that would be needed for this research project. The major bottleneck was video rendering and any videoprocessing was almost impossible, the ARM architecture also made the development difficult. At this stage, the project was viewed as an investigation into how people would respond to the idea of expressive cellphones and how they would want them to operate. To investigate this, we needed a platform that had more computational power, memory and extensibility than most cellphones on the market.

Using a high cost device was justified by the fact that when doing research, many options have to be explored, some of which might not turn out to be interesting as applications. When exploring different ideas, one does not want to be limited by constraints that only need to be imposed before applications get developed for the commercial market.

The range of devices that were considered were the *Ultra Mobile Personal Computers* or UMPCs because of their compactness and usually rich set of peripheral devices and hardware useful for this application (wireless networking, bluetooth, USB, screen, microphone, camera etc.). The UMPC that was chosen was the OQO *Model 02*. This device was chosen mostly because of its light weight and small form factor, its resources were comparable to a few other devices on the market. The OQO 02 ships with *Microsoft Windows XP* and has an *x86* processor which simplifies software development when compared to the *ARM* processor on the N810. A considerable amount of time was spent to set this device up with *Linux* and much progress was made in that direction. This was done in an effort to use a more lightweight operating system and thus have more resources available to the application. It turned out that it was

Figure 2-8: A picture of the MeBot V2 in use.

very difficult to find device drivers that worked with *Linux* for the specific hardware on this device so we decided to simply develop software for the *Windows XP* OS.

Software

An interface was designed to give the operator feedback on the current position and pose of the robot. The interface provided an overhead figure of the robot to view the neck-pan DOF as well as a visualization of the range data. A profile figure of the robot was also provided to view the head-tilt DOF and the arms. For moving the robot around, the operator could click on locations in the overhead map. For controlling the arms, waist and neck of the robot the operator had to drag sliders around.

Since the *GStreamer* libraries that were used for audio/video transmission for V1 are not supported on *Windows*, we had to think about different platforms for media transmission. We had noticed that it looked a little strange to see the face of the operator only occupying a small portion of the device's display in the first version of the robot so we decided to use face detection to locate the operator's face and then only send the face and display it full-screen on the device. This required a very high level of control over the video streaming pipeline and limited our options. We chose to use the *Java Media Framework* (JMF) to transmit audio and designed a custom transport protocol on top of UDP for the video stream. When building this version of the robot, the full capability of JMF was not known to the authors, it was used much more extensively in the fourth prototype of the robot as explained in section 2.5.1.

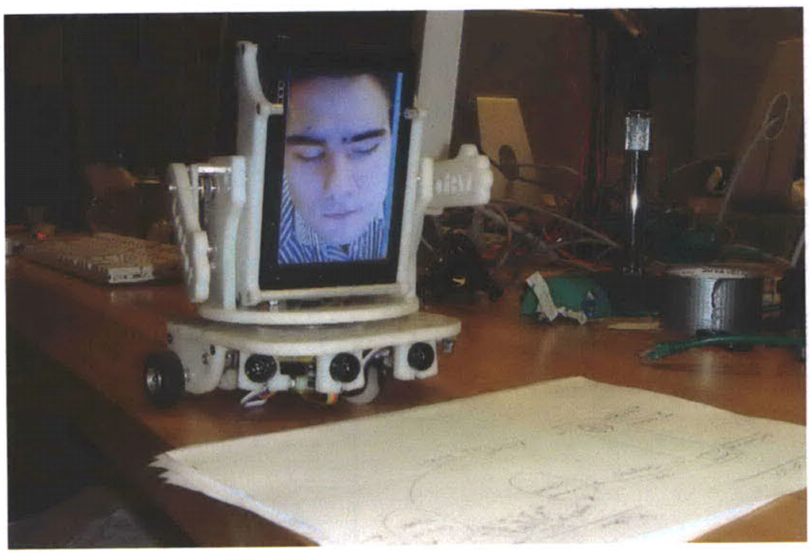

Figure 2-9: A picture of the MeBot V2 in use.

Electronics

Since this version of the robot used DC motors, motor control became more of an issue. We looked for commercially available motor controllers that satisfied our requirements of a small form-factor, a scalable design, choice of potentiometer or encoder control etc. and didn't find an acceptable solution. We started developing our own motor controllers and the result became the first version of the *MCB Mini* motor control scheme which is further explained in section 2.4.2.

This robot had five range sensors, two aiming straight forward, two aiming forward at about 45° angles and one in the back aiming backwards. The range sensors chosen were the *Maxbotics* EZ4 because of their small package, narrow beam width and easy-to-use interface.

When testing the MeBot V1 we noticed that it was difficult for the operator to navigate the robot while maintaining a conversation, especially because operators were afraid that they might drive the robot off the edge of the table. We decided to add sensors to the base of the robot that would detect if there was a surface below the robot and stop the robot if there was none and report it to the operator. The sensor that was chosen was the GP2Y0D805Z0F from *SHARP Microelectronics*. This sensor has an IR LED and a detector, when the IR light has to travel more than 5 *cm* to be reflected back to the IR detector, the sensor reports a changed binary state.

A sensor data acquisition board was designed that polled data from all five range sensors as well as the four surface detectors, serialized it and reported it to the operator. The layout of the sensors can be seen in figure 2-13, this layout was the same for versions two and three.

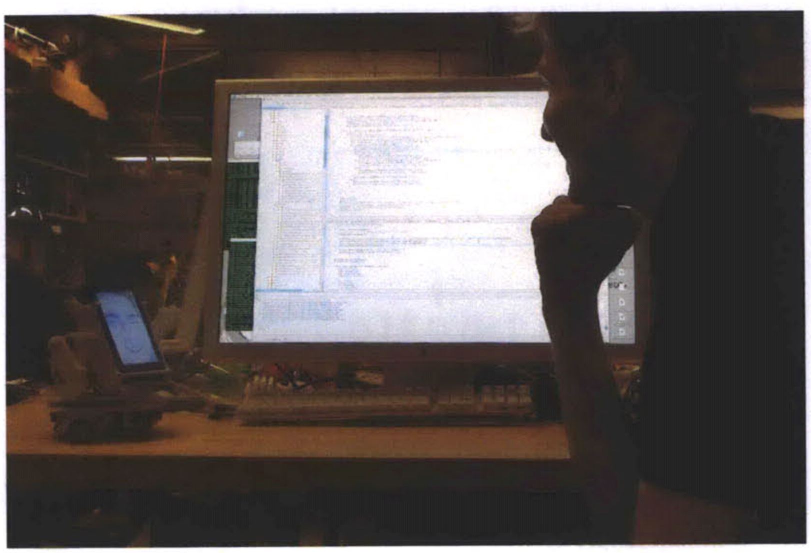

Figure 2-10: A picture of the MeBot V2.

Lessons Learned

The robot was demonstrated during a sponsor week in the Media Lab where we got a lot of good feedback from sponsors and other visitors. While building the robot and designing the systems that supported it we found things that we wanted to improve in future versions of the robot. These things are summarized in the following list:

- One of the bigger criticisms about the robot, had to do with its size or proportions, people felt that the proportion of the head to the rest of the body was a bit off and that the head should be elevated higher. The design of the robot had focused on minimizing its size for portability reasons but this was a very valid point.

- The neck panning DOF gave the robot a bit of a stiff personality as the shoulders and arms of the robot would always move with the neck as it rotated. It was suggested that the head should be able to rotate freely from the arms of the robot.

- The custom video protocol that was designed worked well most of the time but proved to be brittle when the wireless network bandwidth was reduced. It did not provide very good synchronization with the audio. We found that it would be better to find a mature media platform that handles streaming media and provides deep control of the data.

- The motor control update rate was too slow and had to be improved through better design of the communications between the boards.

- Some people felt that the arms were unnecessary if they would only provide one degree of movement. For the use of arms to be valid at all, they would have to be fairly more expressive.

2.2.4 Third Design

After getting a lot of great feedback and generally an excited response from sponsors about the MeBot V2 we started working on the design of the next iteration of the robot.

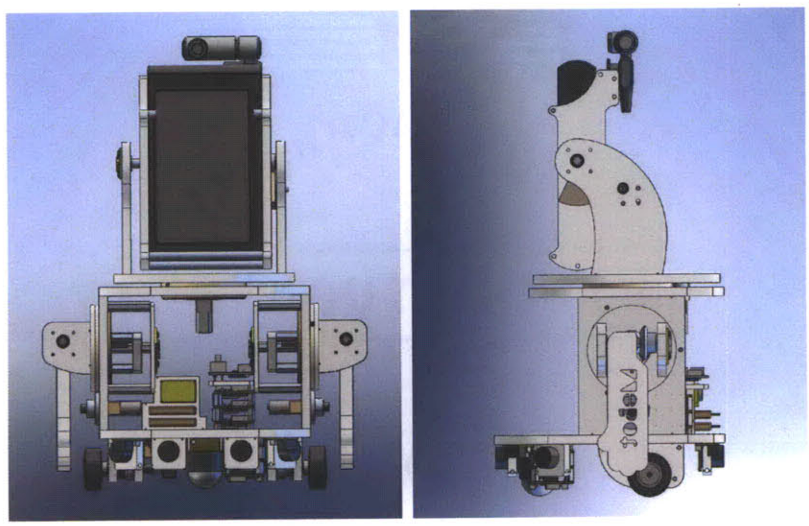

Figure 2-11: A rendering from the mechanical design file for the MeBot V3. Front and side views.

Mechanical design

For this version we wanted to give the operator more freedom in arm movement so that almost arbitrary pointing directions could be achieved without moving the whole robot. We also wanted to enable some expressive gestures such as symbolic clapping or pointing. This was accomplished by changing the shoulder rotation mechanism from a direct 0.25" shaft to a rotation plate with a thrust bearing. The rotation plate was used to mount a GM motor that transmitted torque to the shoulder-extend mechanism via two bevel gears engaged at 90°. This mechanism allowed the robot to rotate the shoulder in its side plane as well as extend the arm out at any angle, giving it an almost arbitrary pointing range as well as capability for more expression.

We felt that the head rotation of the previous version was a bit stiff as it moved the shoulders and arms as well as the head. This effect made it look more like a waist movement than a neck movement. We also felt that the robot needed slightly different proportions, the OQO device was considerably larger than the N810 which made MeBot V2 almost look a little bit like a mobile head with arms as opposed to it resembling a figure that had a head. We decided to solve both of these problems by moving the neck rotation up above the arms of the robot. The increased size of the shoulder mechanism and the elevation of the neck rotation now gave the robot a more elongated and more anthropomorphic look.

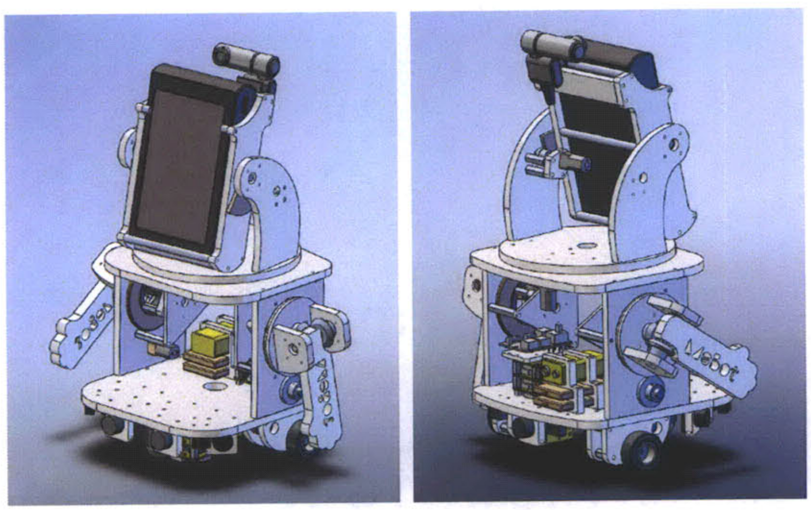

Figure 2-12: A rendering from the mechanical design file for the MeBot V3. Perspective views.

Feedback

When the mechanical design was finished, we decided to have a big design meeting, bring in other students from our group and discuss this prototype's design before going through the time consuming process of building it. Dan Stiehl, Ryan Wistort, Kristopher Dos Santos and Amy Qian attended the meeting and the author thanks them for their valuable input. The general discussion during the meeting mostly focused on issues of stability; the center of mass of the robot had been elevated from the previous version while the base support had not been increased. People also did not like the arms very much and thought they were too far down from the head and possibly too short.

We concluded that this design had possibly been too much of a retrofit to the previous design and possibly a start from scratch would be good at this point, of course making use of lessons learned from previous designs while getting rid of their "ghosts" so to speak.

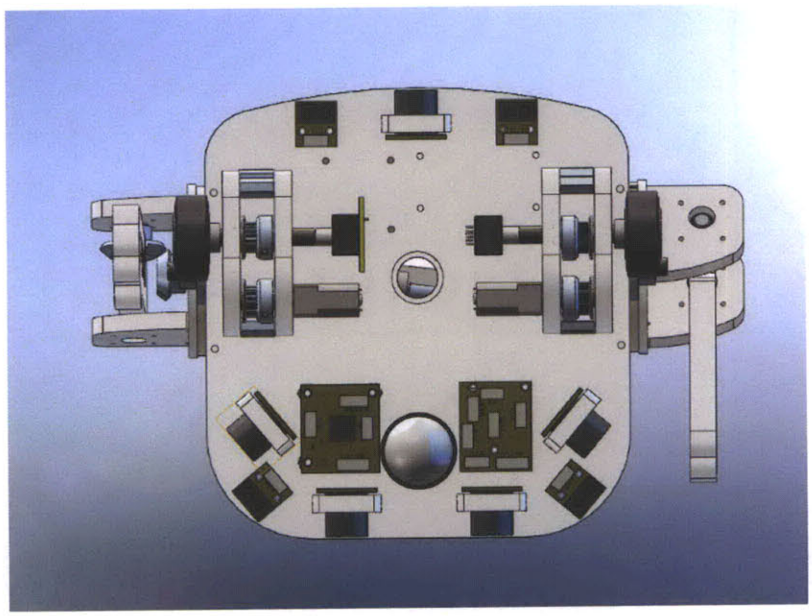

Figure 2-13: A rendering from the mechanical design file for the MeBot V3. Bottom view. To the left of the castor wheel is the sensor acquisition board, to its right is a stack of *MCB Mini* controllers. In the four corners of the base are IR surface detectors and the arrangement of sonar sensors can also be seen.

2.2.5 Fourth Prototype

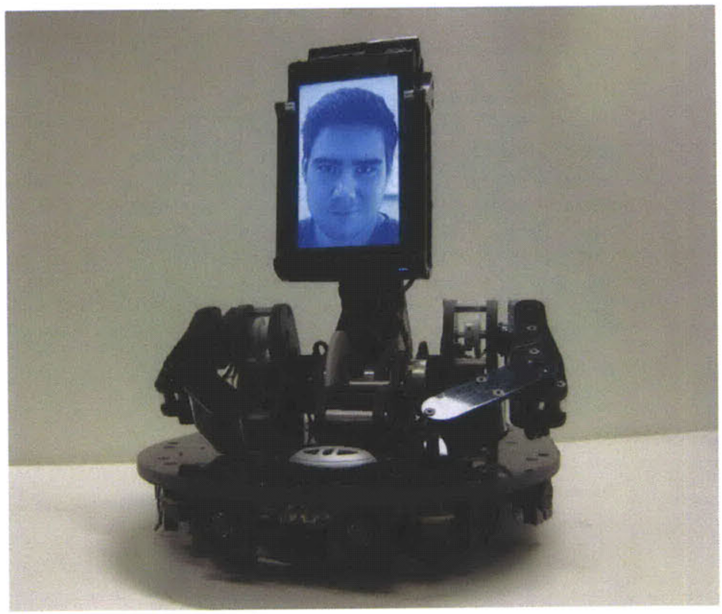

Figure 2-14: A picture that shows a front view of the MeBot V4 (The blue tone is an artifact of the still image, the colors are in fact correct).

Introduction

As was stated in the previous chapter, we wanted to do a complete redesign of the robot, take into account lessons learned and reach further towards the goals that were discussed in the first sections of this chapter. To summarize, we wanted a robot that was able to accommodate expression in gesture, expression through posture or body language, mobility and navigation while still being fairly lightweight and portable.

Mechanical Design

We decided to keep using the human-form analogy as it might be the easiest to understand for people interacting with the robot-mediated operator as well as allowing for fairly direct mapping from control interfaces to the actual movement of the robot.

We wanted to allow more expression in this most recent prototype and we especially wanted to have the robot be able to perform iconic gestures such as waiving, clapping, pointing at self or anything else, etc. To achieve this we decided that the robot arms should have three DOFs, one for shoulder rotation, one for shoulder extension and one for elbow extension.

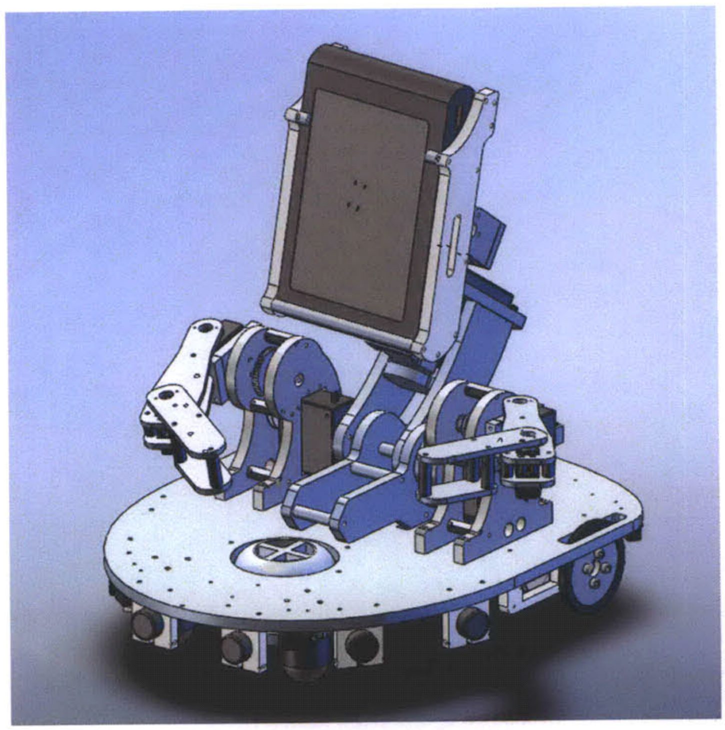

Figure 2-15: A rendering from the mechanical design file for the MeBot V4. A front perspective view.

In this redesign of the robot we decided to completely change the concept of the neck as it had been implemented in previous prototypes. We decided to separate it from the shoulders so that the robot could move the head independently from the arms, we also decided to add a DOF to the neck making it a total of three DOFs. This would give the operator more freedom to control the pose of the robot and convey body language. We thought that allowing the head to move forward and backward in addition to pan and tilt would facilitate the expressions of timidness, shyness, aggressiveness, excitement, curiosity etc.

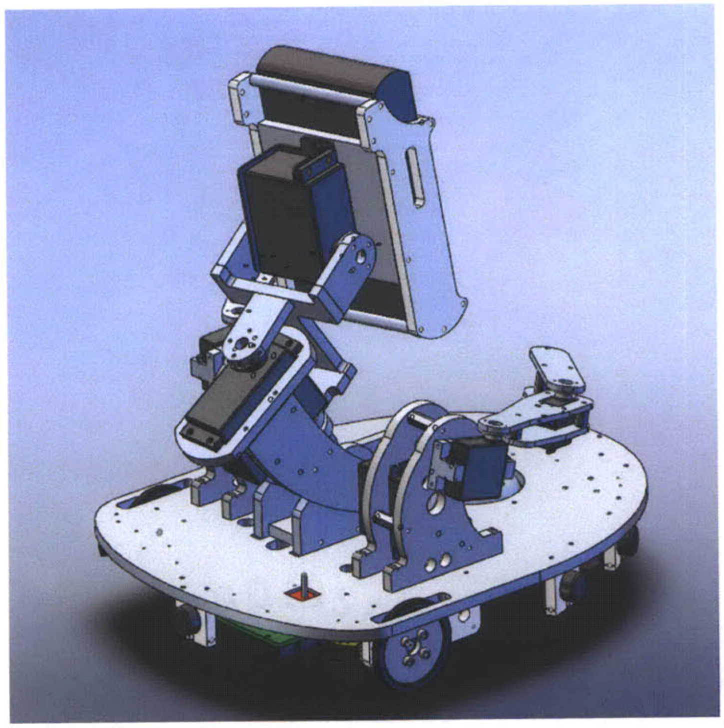

Figure 2-16: A rendering from the mechanical design file for the MeBot V4. A back perspective view.

The design of the neck required some thought because the OQO device, which acts as the head of the robot, contributes most of the weight and it needed to be extended and moved at the tip of the three DOF manipulator that is the neck. This posed a serious cantilever problem and a stability problem, as well as a risk of burning out motors under static load or at the very least generate annoying humming sounds while not in motion. In the first V4 design, digital servos were used for the neck using standard aluminum servo mounts. This turned out to be an unstable mechanism, prone to oscillation. The digital servo motors also generated a lot of noise which rendered that design useless as audio clarity was one of the major design goals. In the second iteration, custom mounts for the motors were built using 0.25" delrin which provided good support and increased stability. Motors with less power density were chosen and the requirement for torque reduced by adding springs that helped support the neck structure under the load of the OQO device. The springs can be seen in figure 2-23, springs were also added to the head-tilt DOF.

The size of the robot's base had to be increased because of the increased range in motion of the head. As was stated earlier, the head was the heaviest part of the robot and as it moved it could obtain quite a lot of inertia, the base of the robot needed to provide support for countering that inertia. A second caster wheel was added so the base would have four points of contact with the ground, providing better support. With a bigger base, all of the electronics and batteries could now be concealed under the base of the robot, this gave the robot a cleaner look. Thought was given to adding a "skirt" to the edge of the base so the electronics would be completely hidden but this was not implemented.

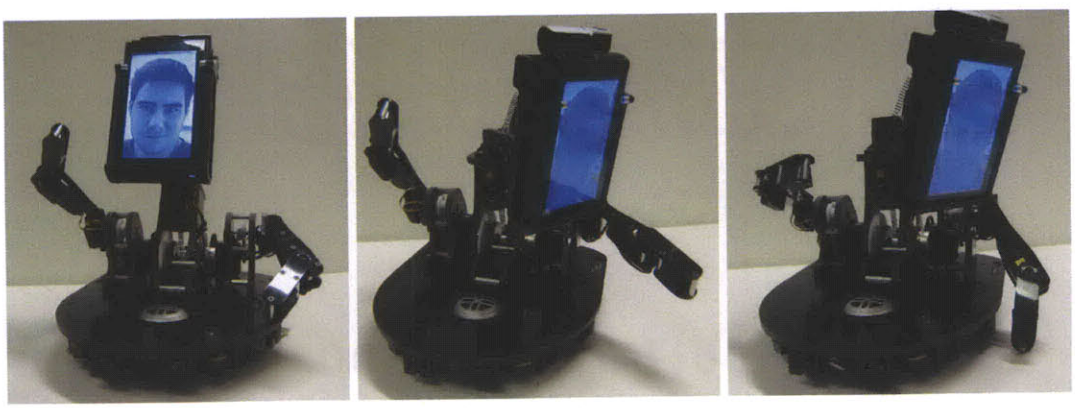

Figure 2-17: A picture that shows multiple views of the MeBot V4 (The blue tone is an artifact of the still image, the colors are in fact correct).

Electronics

A motor control scheme, *MCB Mini*, was designed for the MeBot platform. The design of the motor control scheme was in its second revision when the evaluation of the robot had to take place. Since *MCB Mini V2* was not completely ready by that time, and *MCB Mini V1* could not handle the update rate needed for MeBot V4, we decided to use servo motors and servo controllers for this version of the robot. This was obviously a big compromise to make since much time and effort had been put into the development of the *MCB Mini*. It was nevertheless necessary to finish the evaluation of the robot in time for the deadline. The design of *MCB Mini* is described in detail in section 2.4.

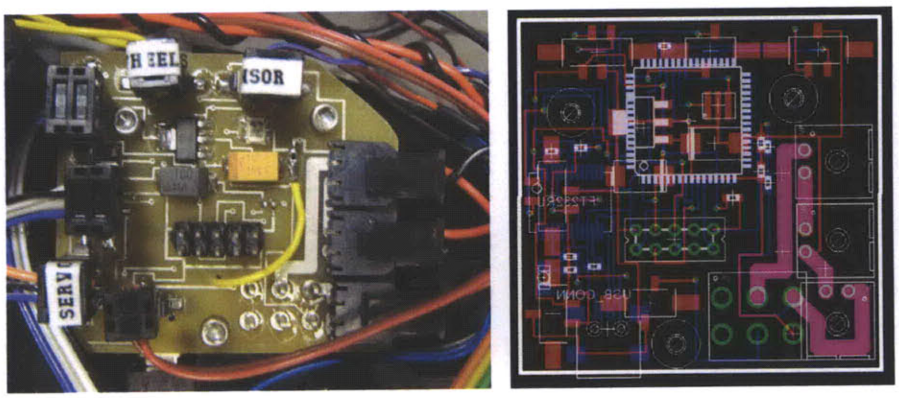

Figure 2-18: This figure shows on the left a picture of the master board in place in the MeBot V4. On the right is a rendering from the *EaglePCB* design file for the same board.

The electronics onboard the MeBot V4 consists of the master board, the sensor board, the servo controllers, servo motors, encoder sensors, range sensor and IR edge-detection sensors. The master board handles all of the low level intelligence of the robot, the sensor board continuously polls data from the range and IR sensors and the servo controllers listen for commands and continuously send control signals to the servo motors. The organization of data for this version of the robot can be viewed in figure 2-20.

The master board performs multiple tasks at once. It monitors the battery voltage for both the logic battery (9V) and motor batteries (6V) and reports them back to the OQO device. It listens for packages coming in from the sensor board and assembles a larger information package to be sent to the OQO. It listens to motor commands coming from the OQO, copies the commands for the wheel motors into memory and

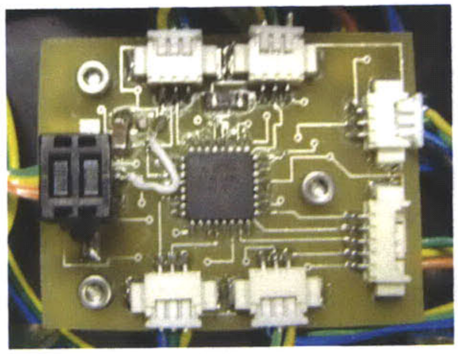

Figure 2-19: A picture of the sensor acquisition board in place in the MeBot V4.

forwards the rest to the servo controllers. The most complicated task it performs is, while continuously monitoring the output of two optical quadrature encoders and integrating their output, it performs a tight PID control loop for both wheels. The loop runs at about 500 Hz and compares the current position of the wheels as reported by the encoders with the commanded position from the OQO. It then asserts a control value to the motors according to a weighted sum of the difference, rate of difference and accumulated difference. The process of PID control is further explained in section 2.4.4. The information package that is assembled and sent to the OQO contains the range and IR data from the sensor board, the encoder tick counts for both wheels and the battery voltages.

In an effort to minimize the cognitive load on the operator while controlling the robot and managing an interaction, we fitted the robot with multiple range sensors. The range sensors help operators understand their remote surroundings without having to look around very much by moving a relatively narrow field-of-view camera. The range information can also enable autonomous navigation and mobility behaviors, this possibility is discussed further in section 4.2. The placement of the range sensors can be seen in figure 2-21, there are two sensors aiming straight ahead, two sensors on each side aiming ahead at different angles and one in the back aiming backwards.

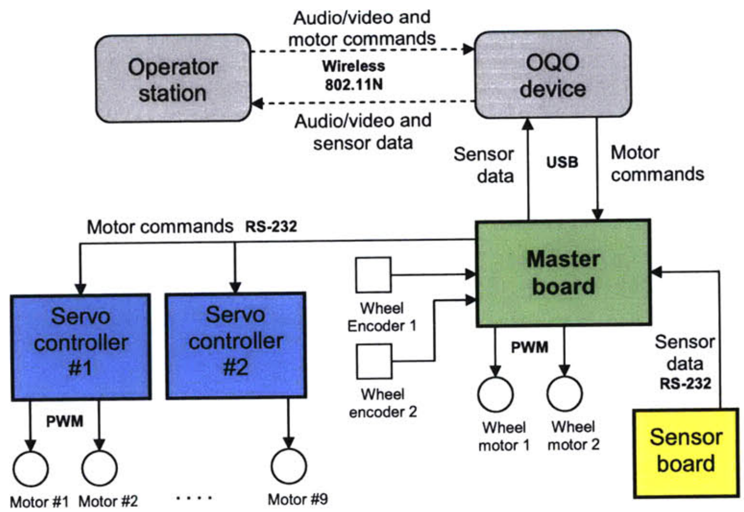

Figure 2-20: This figure shows how data is passed through the robot system. The master board sits at the center of the low-level communication stack and arbitrates much of the data as well as performing control of the wheels.

In further effort to allow the operator to focus on the interaction, we decided to put IR sensors in multiple locations around the robot's base. These sensors detect the presence of a surface under the base. This helps the robot to automatically detect the edge of a desk and not drive off it. The arrangement of the IR sensors can be seen in figure 2-21 but they were placed behind each wheel and in front of each castor wheel.

Figure 2-21 also shows the placement of the electronic boards, in the front is the sensor acquisition board as most of the sensors are in the front anyway. In the back the master board is on one side of the robot and the two servo controllers on the other side. The encoder-servomotor assembly can also be seen in the wheel modules in the same figure.

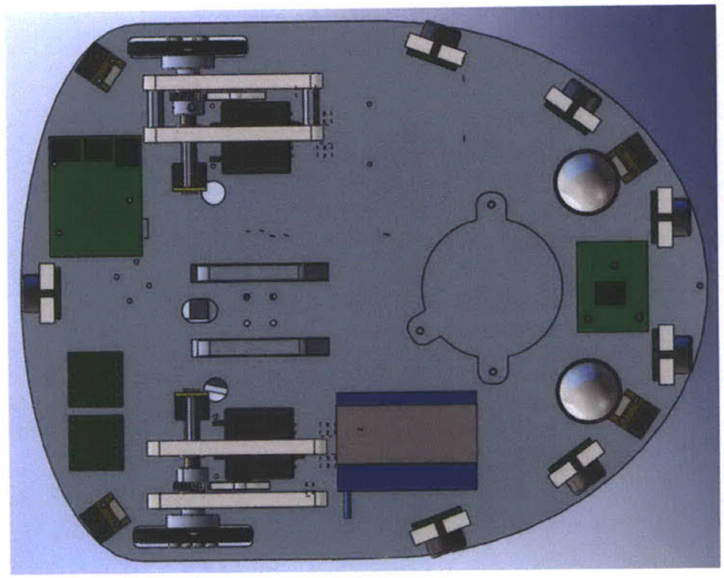

Figure 2-21: A rendering from the mechanical design file for the MeBot V4. A view under the robot.

Software

The interfaces that were designed for this version of the system are described in section 2.3, the audio/video transmission is described in section 2.5.1 and the motor control programming in section 2.4.3.

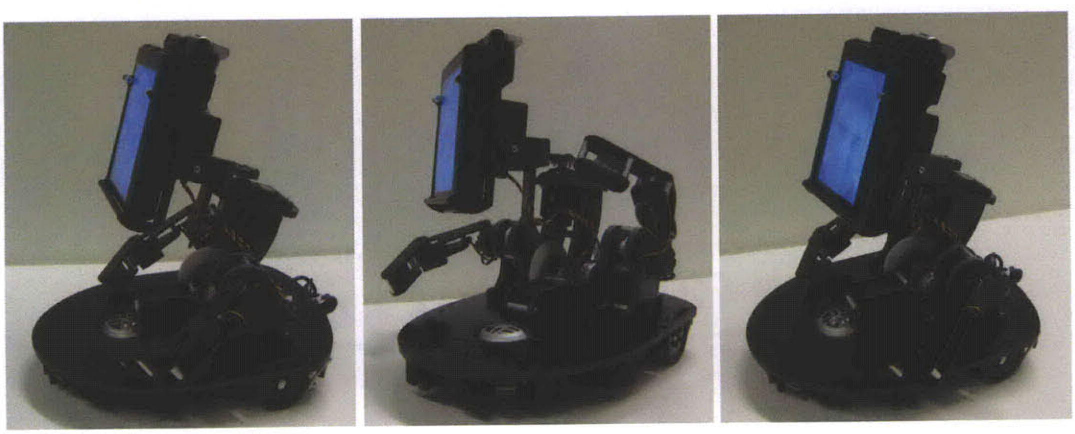

Figure 2-22: A picture that shows multiple views of the MeBot V4 (The blue tone is an artifact of the still image, the colors are in fact correct).

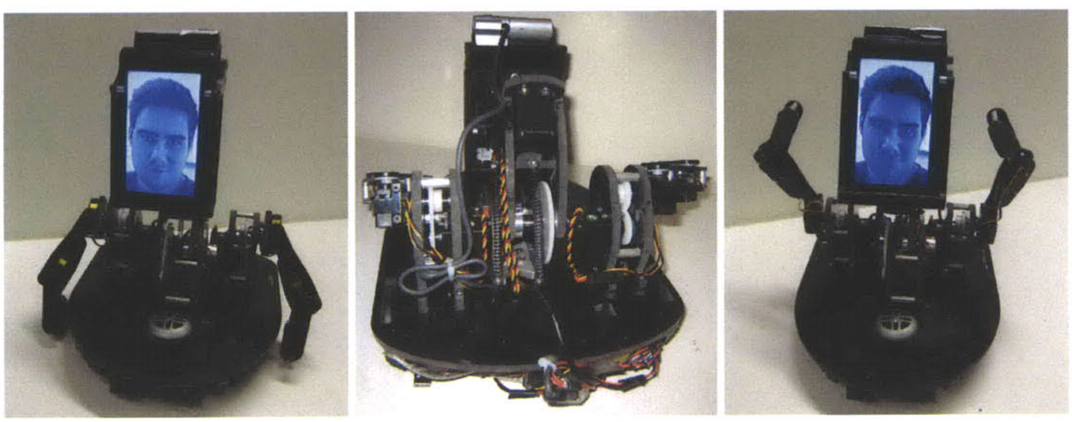

Figure 2-23: A picture that shows multiple views of the MeBot V4 (The blue tone is an artifact of the still image, the colors are in fact correct).

2.3 Interface Design

2.3.1 Specifications

Immersion

It is extremely important, in the applications that the MeBot is intended to serve, that the operator feels immersed in the remote surroundings and that their experience of controlling the robot as well as taking part in the interaction is enjoyable. To allow the operator to feel engaged in the interaction the robot and its interface should provide relatively high quality video and audio data from the remote scene. Other sensors could be employed to help give the operator a deeper understanding of the remote surroundings such as range sensors, directional microphones, pyroelectric motion detectors etc.

Intuitive control

Controlling a robot remotely is usually a very difficult task and it can be hard to perform while trying to maintain a conversation. It is very important in the design of a telepresence system like the MeBot to pay great attention to making the control interfaces as easy to use as possible so as to minimize the cognitive load on the operator. This will allow them to focus on the interaction. It would be great if the operator's instinctive responses could be utilized for control. For example, when a person wants to see something that is at the periphery of their vision, they simply turn their head. If we can sense the operator's head movement and have it control the robot's camera then that would be a degree of control that the operator doesn't have to think too much about. Any low level autonomy that could help reduce the cognitive load of operation must also be employed.

Scalability

Just as the MeBot telepresence robot will be used in many different situations at different locations, the operator of the robot could also be in different locations which might afford them different levels of focus, attention or simply space to control the robot. The control interface needs to consider this and provide multiple ways of controlling the robot that require different levels of engagement and therefore give different levels of depth of control. In an extreme case, the operator should be able to control the robot from a mobile device or a cellphone. Obviously in this mode, the robot would be executing a lot of scripted or autonomous behavior as the operators don't have very rich control modalities available to them nor do they have the full focus required for complete control. The other extreme is when the operator is at their operator station and wants full control of the robot, this could be beneficial to achieving correct communication in critical meeting situations.

2.3.2 Direct Control

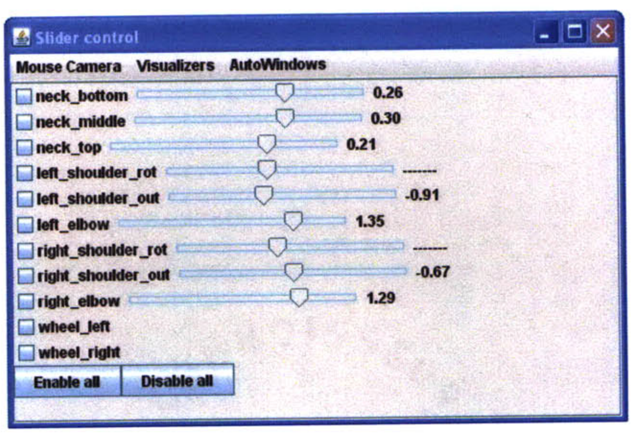

Figure 2-24: A screenshot of the slider control dialog. Here the current motor positions can always be viewed as well as individually manipulated.

The direct control interface was the first one to be built. It allows the operator to individually control every motor in the system by dragging sliders. The slider provides information about the angle position of that particular motor in radians. This interface also provides the option to enable or disable motors. This ability can be very useful, an operator might for example want to make sure that the robot will not move around; he can simply disable the wheel motors.

This display is updated with information provided by other interfaces so it also has great value as a debugging interface. It directly shows the operator or developer the motor position values that the interface being used or analyzed is publishing. For debugging purposes, one can disable all motors and play with the interfaces simply to see how the position values change in the slider interface.

2.3.3 Graphical Interface

The direct interface has its advantages but is hardly the best way to control multiple DOFs in an expressive way. We designed a graphical interface to control the arms and neck of the robot which we thought could be a more intuitive way of controlling the robot rather than dragging around sliders for individual motors.

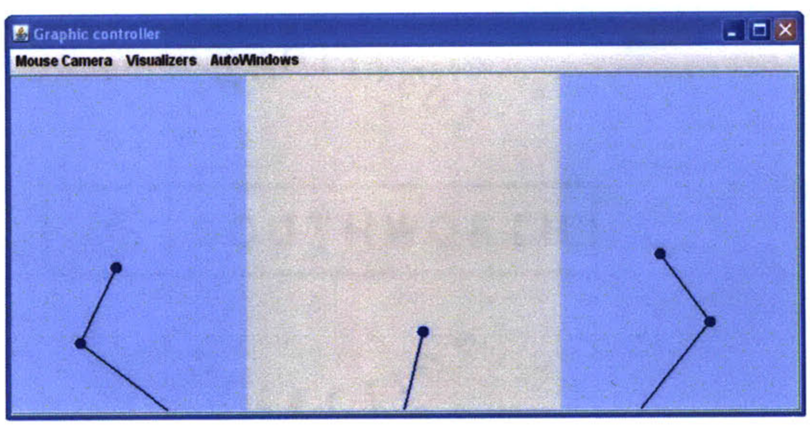

Figure 2-25: A screenshot of the graphical control interface.

The interface can be viewed in figure 2-25. It consists of three appendages representing the arms and neck. The appendages have blue balls at their ends, these balls can be dragged around with the mouse. When the arm appendage is dragged, it needs to satisfy the constraints that the upper and lower "bones" of the arms do not change in size and are always joined and that the shoulder-point of the arm (where the appendage meets the lower boundary of the interface) does not move. The angle between the upper and lower bones is constrained to be in the interval of $\theta \in [0, \pi]$. The joint angles of the arm are determined by performing inverse kinematic (IK) calculations for the two-bar-linkage mechanism that is the arm. The calculations use the mouse-dragging point as a desired destination of the end effector and search for an acceptable solution within the constraints of the physical dimensions of the robot. The length of the neck "bone" can extend and that represents the forward motion of the neck, the range of the extension is obviously also limited by physical dimensions of the robot.

The angle between the base line of the interface and the upper arm "bone" maps to the shoulder-extend DOF. The angle between the upper and lower "bones" maps to the elbow-extend DOFs. The length of the neck "bone" maps to the neck-extend DOF and the angle between the neck "bone" and the baseline of the interface maps to the head-pan DOF. For controlling the shoulder-rotate or head-tilt DOFs the operator needs to right-click with the mouse in any of the colored regions (light blue on the left for the left shoulder, light grey for neck tilt and light blue on the right for the right shoulder) and drag the mouse up and down. The y-factor of the length of the drag in the regions maps to the movement of the corresponding DOF.

2.3.4 Head Movement Control

As was stated in the specifications of the interface design, we wanted to provide the operator with very intuitive ways of controlling the robot so that they could focus their attention on the interaction. We felt that graphical interfaces were a good way to control the robot for many situations but that they were still not the most intuitive method of control. We started looking for methods of intuitive control and came across a company called *Seeing Machines* that has developed a software library called *FaceAPI*. This library uses a video stream from a webcam and finds human faces in the images. When it has found a face, it estimates the 3D location of the face in the scene as well as its pose (orientations). We got a research license for their software and developed an application that reports these values directly to our control interface and provides a way to zero the data when needed (that is, tell the robot that our current head-pose should be its idle pose).

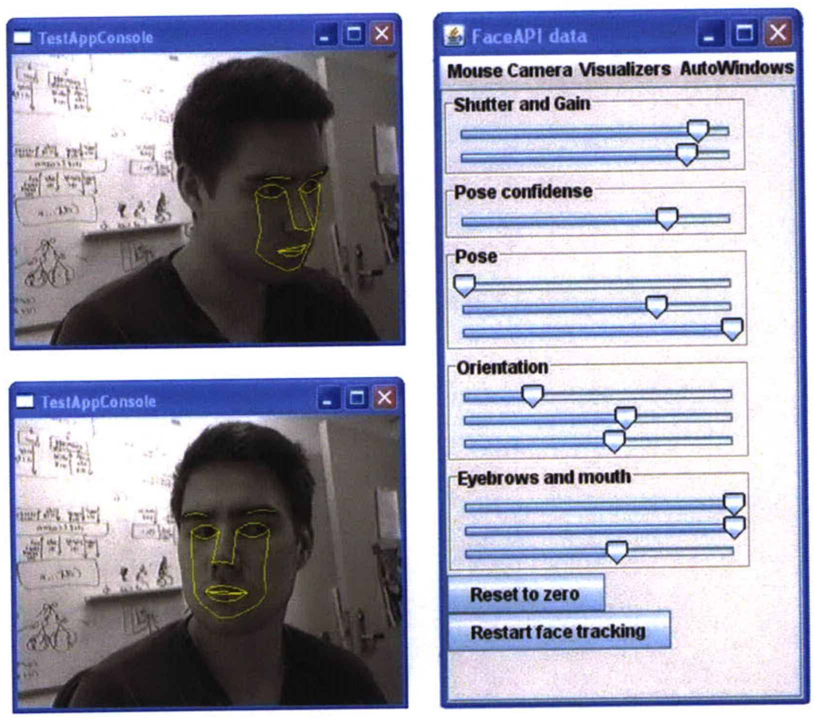

Figure 2-26: A screenshot of the FaceAPI view. The figures on the left show the webcam view from the API and the dialog on the right visualizes the head-pose and orientation data as well as provide some control.

The z-depth of the head was mapped to the neck-extend DOF, the y-rotation was mapped to the neck-pan DOF and the x-rotation was mapped to the neck-tilt DOF. This proved to be an easy way to control the robot's head without having to operate any sort of device, simply by moving your own head. Figure 2-26 shows sample video output from the API as well as the interface we wrote to communicate with it.

2.3.5 Sympathetic Control

We were very pleased with the head-pose estimation control of the robot's neck and how that modality of control was natural and easy to use. We wanted to have the same success in the design of a control interface for the arms of the robot. Controlling the arms of the robot directly with movement of one's own arms is a difficult task to manage as the operator's arms are used for many things other than just controlling the robot. We thought of different ways to achieve this level of control and experimented with capture systems such as inertial measurement units (IMUs) with magnetometers and optical tracking by a *Vicon* system. These methods both required the operator to wear either electronic devices on their body or passive reflective marker badges. It was also difficult to manage the boundary between operating the robot and doing something locally using your arms that doesn't have anything to do with the interaction.

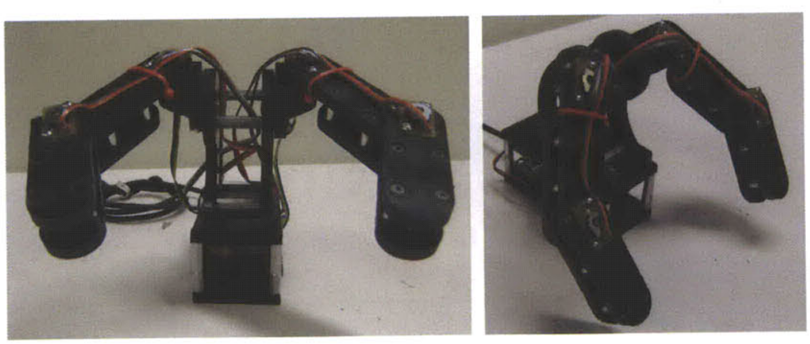

Figure 2-27: A picture that shows a two front views of the sympathetic controller.

We decided that a less intrusive and possibly easier method of control would be to build a sympathetic controller for the arms of the robot. The controller would be a passive model of the robot that had joints in the all the same places and when they were moved, the robot would move in a corresponding way. When the operator lets go of the joints, say to perform a task using their hands that doesn't have anything to do with the interaction, the model simply stays in place and the movement of the robot is not affected. The design of the sympathetic controller was inspired by a similar design to control a robot called The Huggable [Stiehl *et al.*, 2006].

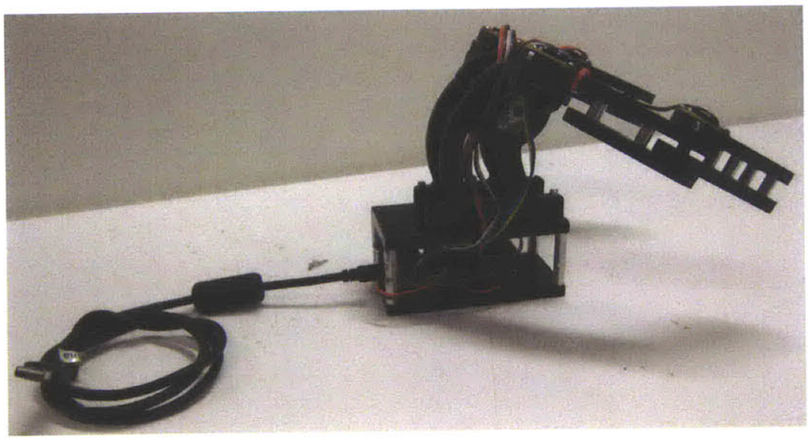

Figure 2-28: A picture that shows a side view of the sympathetic controller.

2.3.6 Visual Feedback

Controlling an eleven DOF robot is generally not an easy task and especially not when it is done at a distance and the operators can not observe the effects of their actions directly. An operator that has never used the robot before or possibly never even seen it will have a very hard time figuring out the different control modalities, given no feedback. We decided to build a 3D *Maya* model of the robot, articulate it in the same way as we do the robot itself, using the motor commands from the interfaces and display it to the operator in real-time. A screenshot of the model can be seen in figure 2-29. This was found to be tremendously useful in training operators as well as for debugging purposes during development as this allowed the development of interfaces without the need for having the robot turned on or even be present.

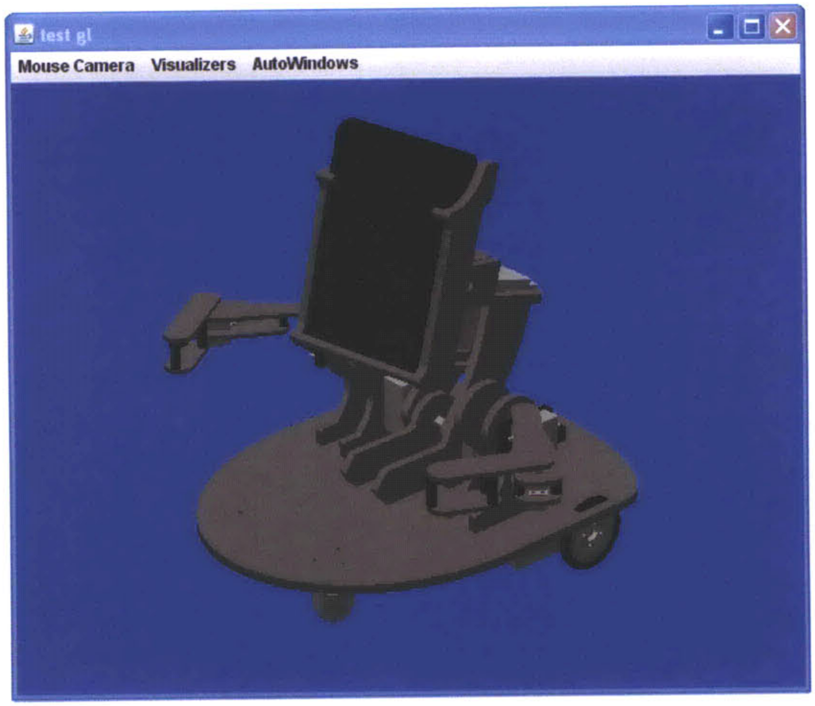

Figure 2-29: A screenshot of the 3D model of the robot as it is presented to the operator.

2.3.7 Space Navigation

For controlling the robot's mobility, a 3D mouse was found to be a good device of control. The 3D mouse offers sensing of movement in six dimensions, translation in x, y and z directions as well as rotation about all of those axes. The mapping between sensed input and robot movement is fairly straight forward, translation of the knob moves a target point in a corresponding manner relative to the robots current location. This is visualized in an overhead graphic that also shows the robot's global position and orientation. Rotation of the knob around the z-axis turns the target location point around the current location of the robot. The operation of the space navigator is explained in figure 2-30

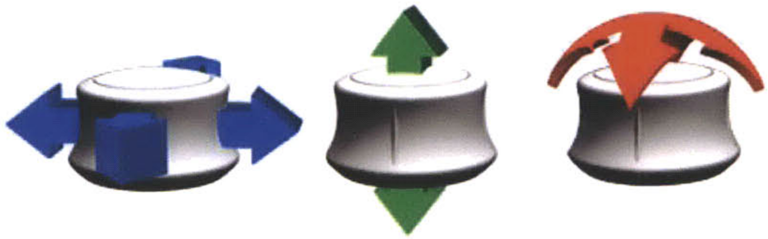

Figure 2-30: Image courtesy of http://3dconnexion.com. A figure explaining the dimensions of translation that the 3D mouse senses, it also senses rotation around the same axes.

2.3.8 Summary

The evaluation of the robot was the first significant test of the interfaces for controlling the fourth version of the robot. During the evaluation, an operator controlled the robot using most of the interfaces that we have designed so far. She would sit in front of the custom display (described in section 2.5.2) where she could observe the remote video. Facing her were two cameras - one embedded in the display for capturing the video feed to be sent to the robot and another camera used for determining her head pose and orientation for controlling the robot's head (2.3.4). She was wearing a headset to listen to the audio coming from the robot, as well as a microphone. She had in front of her the sympathetic controller for operating the arms and shoulders of the robot (2.3.5) as well as the *SpaceNavigator* for controlling the mobility of the robot (2.3.7)

On a secondary display, she had different interfaces available to her. Firstly, she could observe the pose of the robot in a real-time 3D rendering (2.3.6). She could also see the output of the FaceAPI video capture to determine if the head-pose tracking was correct (see figure 2-26). Lastly she had access to two different interfaces for controlling the robot's DOFs if the head tracking and symapthetic would malfunction - the graphical interface control (2.3.3), and the direct control interface (2.3.2).

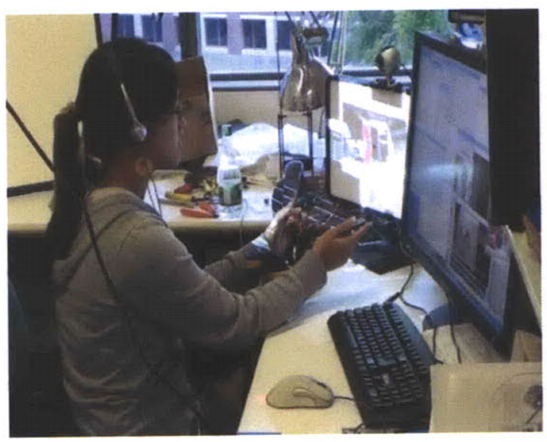

Figure 2-31: The operator had multiple control interfaces available to her.

2.4 MCB Mini: Motor Control Scheme

2.4.1 General Specifications

A motor control scheme was developed to address the very specific needs and demands of the research robots that are developed and used by the Personal Robots group at the Media Lab. The development of robots that should have the ability to display rich expressiveness as well dexterity and movement in human-like ways requires an advanced motor control scheme that fulfills a certain set of requirements. A few of the important specifications for motor control of robots that are developed in the Personal Robots group at the Media Lab are listed as follows:

- **High update rate:** For the robot to move in a compelling and human-like way, the behavior system needs to be able to stream updated target positions for the motors at a decent update rate. The motor boards need to be able to dynamically adapt to a rapid stream of changing target positions as well as possibly compensate by interpolation or filtering for slow update rates resulting from communication lag.

- **High density:** Robots that are intended to display social characteristics usually have rigid constraints on their structure imposed by aesthetics. These constraints usually don't leave a lot of extra space for bulky motor controllers. Robots of this type therefore require high density controllers that can control multiple motors while taking up as little space as possible. A distributed architecture might be useful in this regard as it might be hard to fit all of the motor controllers in one big space but there might be smaller spaces available distributed around the robot.

- **Scalability:** Incremental design is a process of trying out a possible solution to a particular problem and individually evaluating the result through incremental prototyping. This process is particularly useful for designing robots and it requires a scalable motor control system as the number of motors is not a static quantity but can change through the design process. The motor control scheme needs to address this by offering an easy way to add or remove motors.

- **Safe operation:** Sociable robots are essentially designed to interact with people and therefore need to operate safely. Safety can be interpreted in several different ways, the normal operation of the robot needs to be safe for humans and that involves acceleration and velocity limits for joints as well as monitoring of motor currents and responding in a safe way to unexpected values. All failure modes of the robot need to be safe for humans so that if anything goes wrong, the robot will not cause harm. Lastly these robots are usually expensive machines and difficult to repair so the motor control needs to make sure that the robot doesn't do harm to itself.

- **Transparency:** The design of complicated robotic systems usually involves a great deal of debugging. When implementing and testing out new systems, much of the development time goes into trying to understand the source of possibly erroneous behaviors of the system. It is very important that every stage of the robotic system be as transparent as possible, that is to say that communication of signals can be monitored by the designer at every stage so they can determine the origins of the error.

- **Minimal wiring:** Sociable robots usually have a very dense network of relatively low-powered motors. These motors are placed in various locations all over the robot and can be hard to access and especially tricky to trace wires to. The motor control scheme must be implemented so that it requires minimal wiring and an easy way to switch motors or boards out if they become unusable through wear-and-tear.

2.4.2 MCB Mini V1

A very capable motor control scheme was designed in our group by Matt Hancher [Hancher, 2003]. This system fulfills most requirements listed above but it was considered too bulky and had too many channels for our initial designs. That is why we decided to go ahead and design a more scalable solution intended for smaller platforms.

The *MCB Mini V1* boards were designed to use the *Freescale Semiconductor* MC33887 H-bridge. This chip provides a MOSFET H-bridge as well as gate drivers with control logic. The chip also provides a current feedback signal. The design used an *AVR ATmega88* microcontroller to implement the PID controller, PID control is explained in section 2.4.4.

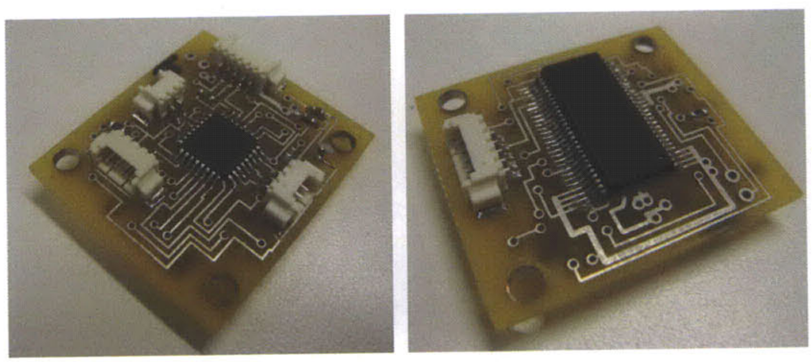

Figure 2-32: Both sides of the motor control board for *MCB Mini V1*.

In this implementation, the master board didn't serve any purpose other than to bridge messages between the PC and the motor control boards, that is to say that it didn't contain any intelligence and in fact not even a microcontroller. Two different master boards were designed for this version to allow for two different transports, a USB board that used an FTDI chip as a bridge and a bluetooth board that used the RN-41 chip from *Roving Networks*.

All motor control boards could control one motor and listened to serial messages from a bus. The serial messages adhered to the RS-232 protocol at TTL levels. Once a board recognized a new package on the bus that contained its ID, it would transfer the contents of the package into its memory and send its sensor data back on the bus. The bus that was meant for communication from the motor controller boards to the master was a high impedance bus with a pull-down resistor to set the default idle bus level to 0V. After sending a package, the host (usually a PC) would wait for a response from that particular motor controller. The motor controller would acquire control of the bus once it recognized its ID and release control when it was done sending its data.

The *MCB Mini V1* was used in MeBot V2 but its update rate of the was found to be too slow and also the bus scheme implemented didn't have enough protection from noise.

2.4.3 MCB Mini V2

In response to the problems we experienced with the first version of the motor control scheme, we decided to do another iteration on our design. We found an H-bridge IC that provided more outputs than the MC33887. The *ST Electronics* L298 IC is a dual H-bridge and driver in a package of similar size as the MC33887. Using this IC in our design now allowed the control of two motors per board which reduced wiring and could increase update rates. We also chose to switch to a different bus scheme for this design. Further descriptions are provided in the following sections.

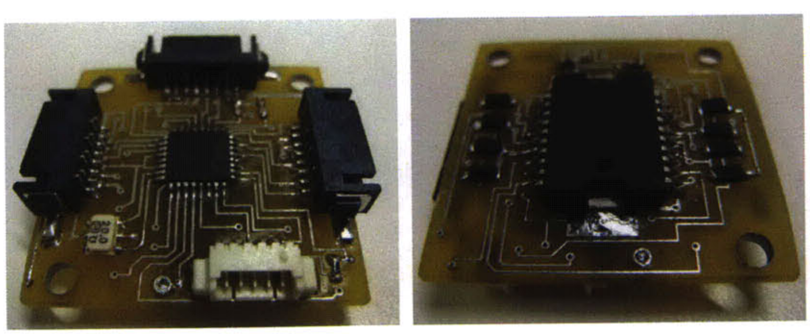

Figure 2-33: The motor control board for *MCB Mini V2.*

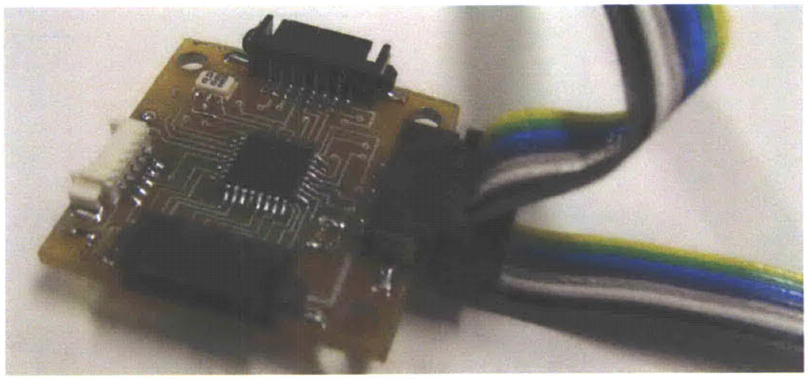

Figure 2-34: The motor control board for *MCB Mini V2*. Figure shows how boards connect simply to a bus that provides data and power.

Communications

As can be seen in figure 2-35, the *MCB Mini* scheme uses a highly layered approach. Using such a layered system has the advantage of being modular and offers the possibility for distributed control. Layering comes at a price, the same data needs to be cached at multiple levels and synchronized safely between them.

Messages Between Behavior System and Motorserver

At the highest level of any robot control scheme there usually resides a *Behavior system* of some sort that uses multiple sensory information as well as user input to decide how the robot should move its joints. This system can be arbitrarily complex and possibly even distributed and there should not be a requirement for it to be written in any specific programming language. It is the view of the author that the motor control scheme should be implemented in such a way that it poses minimal constraints on the programmer who designs the higher level behaviors of the robot, while still trying to accommodate for their needs.

The Personal Robots group has developed a communication protocol called the *Intra-robot communication protocol* (IRCP). This is a thin layer on top of UDP that

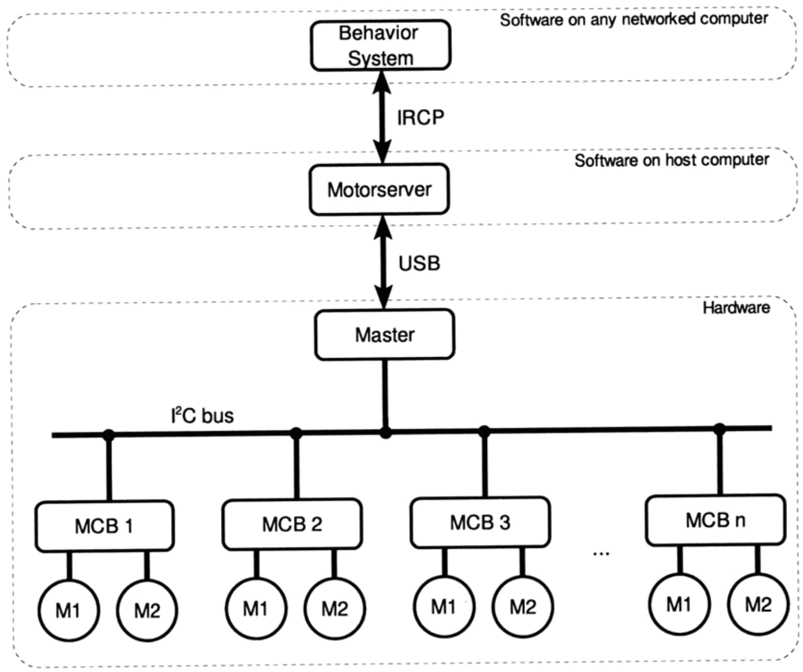

Figure 2-35: A high-level overview of the communication protocols used in the *MCB Mini* motor control scheme.

abstracts away IP addresses and introduces instead robot IDs and module IDs for any particular robot. Any piece of software that runs on a networked computer can broadcast its robot and module ID and subsequently start sending and receiving information to and from other modules on the same network.

In the *MCB Mini* implementation, the motorserver is a module which has a static module ID that corresponds to motor control. This means that any software module that wants to control the robots simply needs to query the motorserver module for the motors that it controls, subscribe to their updated true positions and start sending target positions. Through this interface, many more messages can be received such as debug packages from any specific motor board containing much of its internal data for debugging purposes, status messages from the master, error messages from motor boards or the master, updated parameters to any motor board etc.

Messages Between Motorserver and Master

The motorserver needs to run on a computer that has a wired USB connection to the master. Both the master and the motorserver receive and send information using the RS-232 serial protocol using TTL levels and a fast baud rate. The serial messages are converted to USB to travel across the physical link by an FTDI device. This device fully encapsulates the USB stack and emulates a UART port.

The master is responsible for receiving information from the motorserver and relay it onward to the motor boards. The master and motorserver communicate in a synchronous way, that is the master waits for a package from the motorserver, that usually contains a target position update message, and when it has received it the master will respond by sending a package back that contains a current position message.

The master handles all this communication with the motorserver in interrupt routines. This means that it is running its main loop continuously but responds to incoming motorserver bytes simply by putting them in a buffer and when the package is ready it will service the package and send back a response package. In its main loop the master simply updates target positions to all boards and polls them for current positions as fast as it can on the I^2C bus. If the master has been queried for any specific thing like a bus sweep or debug data by the motorserver, it will perform that task in the next update loop. The master keeps track of the time-interval between target position updates from the motorserver and if a predefined number of milliseconds has passed between receiving those messages, the master will determine that something is wrong and disable the boards for safety reasons.

A certain packet structure is used when communicating between the master and motorserver. The packet structure is the same going both directions but the messages that are put in it vary between directions. The general structure can be seen in figure 2-36. Each packet can contain an arbitrary number of messages and its framing

Byte nr.	Content	Description
0	Type of msg. 1	Determines the type of message 1
1	# of msg. bytes	Number of info bytes in message 1
2	Msg. byte 1	Info bytes
3	Msg. byte 2	
	...	
	Last Msg byte	
	Type of msg. 2	Determines the type of message 2
	...	
	Type of msg. n	Determines the type of message n
	...	
	Checksum	Sum of all previous bytes modulo 256
	0x01	Unique post-header
	0x02	to signal end of packet
	0x03	

Figure 2-36: The general packet structure that gets sent from master to motorserver and visa versa. This package can contain any number of different messages.

is determined by a sequence of four bytes containing the numbers one, two, three and four. The byte before this post-header will be a checksum to confirm correct reception of the packet. Various message types can be passed in a packet but their structure is always the same, starting with a message byte to determine what sort of a message this is, followed by a byte that states the number of bytes to follow in this message and finally the information in the message. Regardless of whether there are any messages to pass or if any boards have been initialized, the motorserver will send a packet in every update loop that contains target positions, this could be a message containing zero message bytes if no boards have been initialized. The master will respond with current positions every time regardless of whether any boards are connected to the master. This is important so that the master knows if there is a motorserver connected at all times.

The various messages passed from the motorserver to the master can be seen in figures 2-37, 2-38 and 2-39 and the various messages passed from the master to the

Byte nr.	Content	Description
0	0x00	Target pos. message
1	# of msg. bytes	Should be 9x[# of boards]
2	ID 1	First ID
3-6	Target pos. 1 A	4 bytes, big-endian
7-10	Target pos. 1 B	4 bytes, big-endian
11	ID 2	Second ID
12-16	Target pos. 2 A	4 bytes, big-endian
17-20	Target pos. 2 B	4 bytes, big-endian
	...	
	ID n	Last ID
	Target pos. n A	4 bytes, big-endian
	Target pos. n B	4 bytes, big-endian

Figure 2-37: Target position message from motorserver to master.

Byte nr.	Content	Description
0	0x01	Parameter message
1	# of msg. bytes	Should be 21x[# of boards]
2	ID 1	The ID of the first board
3	Direction A	The direction of channel A (1 or 0)
4-5	P gain A	2 bytes, big-endian
6-7	D gain A	2 bytes, big-endian
8-9	I gain A	2 bytes, big-endian
10	Enable A	Channel enabled (1 or 0)
11	Pot or encoder A	Pot or encoder control (1 or 0)
12	Debug A	Debug mode (1 or 0)
13	Direction B	The same info for channel B
	...	Unique post-header
22	Debug B	to signal end of packet
	...	
	ID n	Last ID
	...	

Figure 2-38: Parameter message from motorserver to master.

Message byte	Description
0x02	Query for debug data from a motor board
0x03	Query for a bus sweep
0x04	Query for a status message

Figure 2-39: Query message from motorserver to master.

Byte nr.	Content	Description
0	0x00	Current pos. message
1	# of msg. bytes	Should be 9x[# of boards]
2	ID 1	First ID
3-6	Current pos. 1 A	4 bytes, big-endian
7-10	Current pos. 1 B	4 bytes, big-endian
11	ID 2	Second ID
12-16	Current pos. 2 A	4 bytes, big-endian
17-20	Current pos. 2 B	4 bytes, big-endian
	...	
	ID n	Last ID
	Current pos. n A	4 bytes, big-endian
	Current pos. n B	4 bytes, big-endian

Figure 2-40: Current position message from master to motorserver.

Byte nr.	Content	Description
0	0x02	Debug message
1	# of msg. bytes	Should be 37x[# of boards]
2	ID 1	First ID
3-6	Current pos. 1 A	4 bytes, big-endian
7-10	Target pos. 1 A	4 bytes, big-endian
11	Direction 1 A	Direction of channel A (1 or 0)
12-13	P gain 1 A	2 bytes, big-endian
14-15	D gain 1 A	2 bytes, big-endian
16-17	I gain 1 A	2 bytes, big-endian
18	Enable 1 A	Channel enabled (1 or 0)
19	Pot or encoder 1 A	Pot or encoder control (1 or 0)
20	Debug 1 A	Debug mode (1 or 0)
21-24	Current pos. 1 B	4 bytes, big-endian
	...	
	ID n	Last ID
	...	

Figure 2-41: Debug information message from master to motorserver.

Byte nr.	Content	Description
0	0x01	Status message
1	# of msg. bytes	Should be 4x[# of boards]
2	ID 1	First ID
3	Send errors 1	# times master failed to send info to board
4	Receive errors 1	# times master failed to receive info from board
5	Parameter sends 1	# times master had to resend parameters to board
	ID 2	Second ID
	...	
	ID 2	Last ID
	...	

Figure 2-42: Status information message from master to motorserver.

Message byte	Description
0x03	Query for id but there are no ids in use
0x04	Query for id but that id is not in use
0x05	Bus sweep, following ids responded
0x06	Attempt to initialize reserved id 0
0x07	Received parameters for board that had been initialize
0x08	Master reports receiving package with wrong checksu
0x09	Failed attempt to set parameters for board
0x0A	Failed attempt to communicate with motor board
0x0B	Failed attempt to get debug data
0x0C	Board got reset
0x0D	Master reports that it an unknown command
0x0E	Timeout in receive cycle, disabled all channels

Figure 2-43: Various other messages from master to motorserver.

motorserver can be seen in figures 2-40, 2-41, 2-42 and 2-43.

Messages Between Master and Slaves

The master communicates with the motorserver in interrupt routines that effectively run in a different thread than the main loop in the firmware. In a normal update loop, the master will simply send updated target positions to each motor board that has been initialized and then it will poll them for their current positions. This happens as fast as the I^2C bus will allow. In the main loop, the master will also check a flag to see if any boards should have their parameters updated and do so if they should.

It will check another flag to see if it should poll any boards for their debug data and do so if appropriate. It will perform a bus sweep if it has been requested, that is try to contact all possible board IDs on the bus and see which ones respond. Lastly it will check to see if it should send a status message to the motorserver which contains information about how responsive the motor boards have been.

I^2C Bus Protocol

The master communicates with the slaves using an I^2C bus as has been previously stated. I^2C is a communication protocol originally invented by *Philips*, it is a multi-master bus with, in its simplest form, a seven bit slave address space which means that there could possibly be up to 126 slaves because the address zero is reserved for broadcasting. On the physical layer, there are two open-drain communication lines, SCL (clock) and SDA (data), these lines are pulled to V_{CC} by resistors whose values are determined by the capacitance of the lines.

The master initiates and controls all signals on the lines. When the master wants to send data to a particular slave, it will set a *start* condition on the bus which will cause all slaves to start listening. Then the master will put eight bits of data onto the bus, the first bit a *read/write* bit and then the seven-bit address of the slave that should respond. The *read/write* bit will tell the slave whether it should receive or send data. Once the master has sent this address byte, it will wait for an *ack* condition on the bus which will be generated by the slave that recognized its ID. Then the master will send as many bytes as it sees fit, and the slave will always send an *ack* after each one. Once the master is done sending all bytes, it will generate a *stop* condition on the bus that will alert the slave that all of the bytes have been sent.

When the master wants to receive data from a slave, it will again send the address byte with the *read/write* bit set appropriately. Then it will release control of the SDA line following an *ack* from the slave. Now the slave will send a data byte onto the bus and then wait for an *ack* from the master. Once the *ack* has been received, the slave can send the next byte and so forth. If the master doesn't respond to a sent byte with an *ack* then that is considered a *nack* condition and it means that the master doesn't expect or want the slave to send more data onto the bus. The master will then put a *stop* condition on the bus and then it is ready for the next data transmission.

Update Rates

Being aware of update rates is important in applications that command the trajectories of motors because if there is ever a slowdown at any level and the update rate gets too low, the motors could start showing choppy behaviors and this could even result in a dangerous situation. In this system there are multiple update rates that need to be considered.

The update loop that is at the lowest possible level in the system is the pulse-width modulation (PWM) update frequency of the motor control boards. The concept of PWM will be further explained in section 2.4.4. This update loop needs to have a base frequency that is out of the audible range for humans, otherwise it is possible for the motors to start giving off a high-pitch humming sound.

The next update rate to be considered is the frequency of the PID update. Again the concept of PID will be further explained in section 2.4.4 but suffice it to say that this update needs happen frequently but it is also very important that it happens steadily. Usually a frequency of about 500 Hz should be good. If this loop is too fast then the D term is likely to be very noisy and D control becomes difficult. If it is too slow, then undersampling the dynamics of how the motor and its load are responding to your control might occur and the control system could become unstable.

An important update frequency to pay attention to is how often the master updates the desired position and polls for current positions of the slaves. This needs to happen fast enough to provide a smooth enough trajectory of motion for the joints of the robot but there is no reason for this update to happen more frequently than incoming messages from the motorserver unless there is some interpolation being performed at this level. In this system there are two types of filters implemented at this level that provide some interpolation between updates, the first one is a running average infinite impulse response (IIR) filter of an arbitrary order and the second one is a maximum increase filter that has an arbitrary step. The first filter smoothes out transitions between desired positions which serves to reduce noise on the power lines. The second filter is a security measure that basically says that the motor should never be told to take a step that is larger than a certain size in any update.

2.4.4 Control Systems

In this chapter I will discuss some of the concepts that have to do with the actual control of the motors. I will briefly describe pulse width modulation as a way to generate an analog voltage signal as well as PID control as a special case of a feedback control system.

Pulse Width Modulation

Pulse-width modulation (PWM) is a way to generate an analog signal using only simple digital components. PWM is a binary-state signal that consists of a pulse train, each pulse starting at a set time-interval from when the last pulse started but lasting variably long depending on the desired amplitude of the analog signal. More formally put, a PWM signal is a signal of a fixed frequency that has a variable duty cycle. This scheme only requires components that can output binary voltage states and are able to switch state fairly quickly.

An example of a PWM signal can be seen in figure 2-44. Usually, a signal of this sort is used to control a system or a *plant* to use the *Control Systems* vocabulary. This plant can be anything that takes an input signal and gives an output effect, examples of plants could be loudspeakers, motors or heating elements. All of the examples mentioned here are "slow", that is to say that there is a limit to how fast

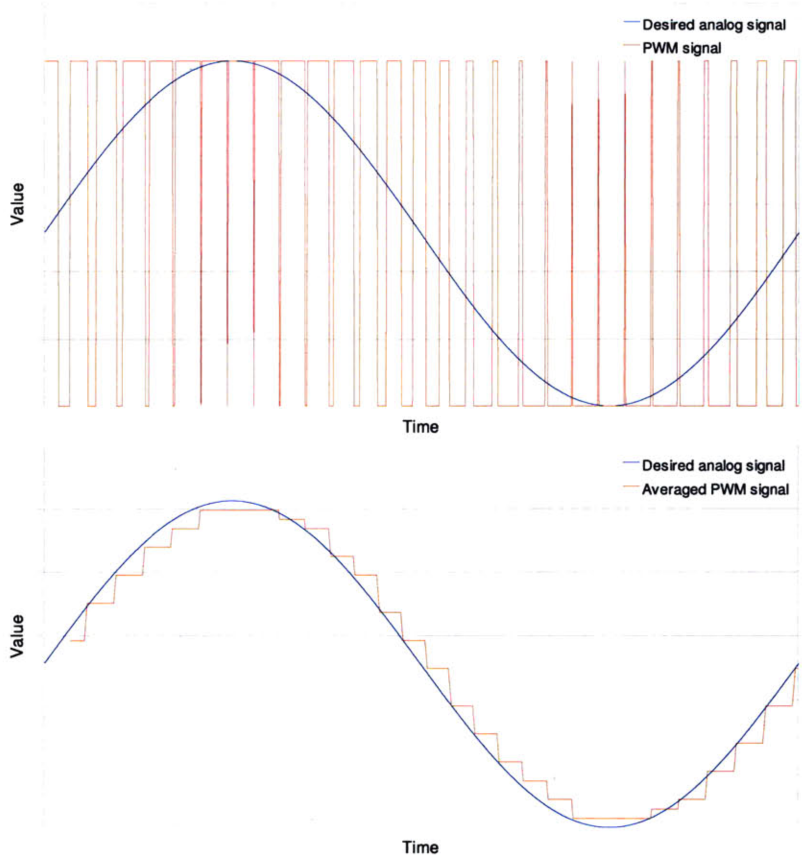

Figure 2-44: The top graph shows a desired analog curve and a PWM signal that approximates that curve. The lower picture shows a filtered version of the PWM signal (uniform coefficient FIR windowed filter). This could represent the average signal seen at some plant that has a low-pass filtering effect on its control signal.

they can respond to their control signal. The loudspeaker has to physically move a diaphragm to set off sound waves, the motor needs to rotate and possibly move a load that has inertia to movement and the heating element has to warm up and warm up its environment. These plants all exhibit a low-pass filtering behavior which is ideal for PWM control since the PWM signal needs to be low-pass filtered anyway to extract the desired curve. This can be seen in the lower graph in figure 2-44.

An important consideration in choosing the parameters of the PWM signal is its fundamental frequency, that is what is the rate at which the pulses get reset and the next duty cycle begins. If analyzed in the frequency domain as opposed to the time domain, the PWM signal would be observed to carry most of its energy at its fundamental frequency and higher frequencies. This becomes important if there are any parts of the system that can resonate with audible frequencies and start essentially "humming". This often happens with motors if their control signals contain audible frequencies. People's hearing responds differently to different frequencies and most adults don't hear sounds that are above 20 kHz in pitch [Caldarelli & Campanella, 2003]. That is why it is important to maintain a fundamental frequency of the PWM signal that is above 20 kHz.

PID Control

Feedback control is a widely accepted way to control many different types of systems. The basic idea is to measure the output of a system, compare that output with a desired output and generate a control signal for the system accordingly. A general diagram of a feedback control system can be viewed in figure 2-45. The main components of this system are the *plant* (this is the system we are trying to control), the *sensor* which measures the output of the plant and the *control system* that generates a control signal according to the difference between the measured output-signal and the desired output.

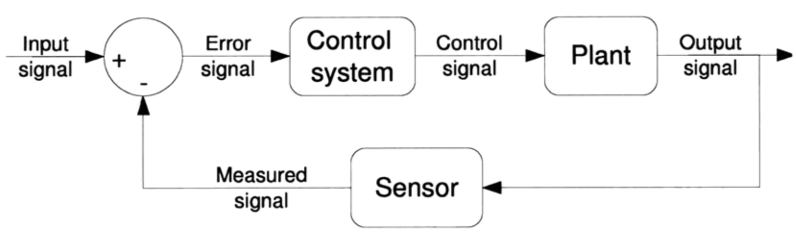

Figure 2-45: This figure shows a classic feedback control scenario. An input signal is provided, the difference between the input and measured output is fed into the control system. The control system finally provides a control signal to the plant that ideally will respond with an output signal that will minimize the difference between the input signal and measured output signal.

PID (Proportional, Integral and Differential) is a certain type of control system in the range of feedback control systems. This particular controller feeds the *error signal* to three different modules in parallel, each module performs some calculations on the signal and produces an output signal, those three output signals are then summed up to produce the *control signal.* The first module simply multiplies the input signal with a constant usually called p_P. The second module feeds the signal through a differentiator and then multiplies with a constant called p_D. The last module integrates the signal and multiplies the output with a constant called p_I.

The basic PID setup can be modified to improve its behavior. For example, if the sensor signal is noisy, the differentiator might simply cause a lot of trouble by amplifying the noise, therefore designers will sometimes put a low-pass filter on the differentiator output. Another useful modification is to put a limit on the integrator so it will only integrate to a certain limit. This can solve problems for a control system that might overreact to a steady impedance that is suddenly removed.

When this type of control system has been put into place, it needs to be calibrated for the particular plant it is meant to control. The calibration is meant to minimize effects such as *ringing*, *overshoot* and *steady state error* which are behaviors that are not desired in a system that should be well controlled. Usually the parameters can be "tweaked" so that the control system works within the constrains that are set for it but there are also analytical methods for determining the constants i.e. the Ziegler-Nichols method [K. J. Åström, 1988]. Note that any of the constants can be set to zero to effectively cancel that particular type of control.

2.5 Auxiliary Systems Design

2.5.1 Real-time Audio/Video

One of the biggest challenges in implementing the MeBot system was creating a real-time audio/video (AV) transmission system that was fast and synchronized while not consuming all of the mobile devices resources so some could be used for sensor processing and motor control. An additional requirement for the AV system was that it provided programmatic access to the video and audio data so that it could be further processed (for example so that face tracking was possible, see section 2.5.1).

Java Media Framework

We evaluated multiple libraries for establishing AV communication between the operator station and the robot, but had trouble with some of them because of any or some of the following reasons: Poor portability between operating systems, lack of codec support for the platform, little or no programmatic access to the data in the media streams or poor *Java* support. We finally came across a framework that seemed to fit our requirements very well, the *Java Media Framework* (JMF).

JMF is an easily extensible media framework for streaming any kind of media between computers, it is also capable of media capture and playback. Its structure defines *datasources* (example: webcam or network port) and *datasinks* (example: video card or IP address and network port). Essentially, a pipeline is set up with a *datasource*, the required format conversions or compression, custom filters and a *datasink*. A fairly easy-to-use interface is provided to write custom filters which satisfy our requirement for access to raw media data.

Face Tracking

From the first prototype of the robot we had learned that if the video of the operator is sent unmodified to the robot, the robot's head won't give a convincing representation of the operator as the operator's face occupies only a portion of the screen and it moves around as the operator moves. This doesn't convey the idea that the robot is representing the operator. We decided that we would have to use face-detection to locate the operator's face in the video and send only the face portion of the video to the device.

We wrote *OpenCV* wrappers for *Java* and embedded the face-detection in the JMF pipeline. Doing this presented a couple of challenges right away, they are listed as follows: A face was not detected in every frame, sometimes faces would be detected where there was no face, if another person walked through the scene *OpenCV* would detect that face instead of the operator's face, the action of searching for faces took a long time, even when the correct face was found the dimensions of it were different between most detections. These problems led to a very choppy video being displayed on the robot and frequently something completely different than the operator was displayed.

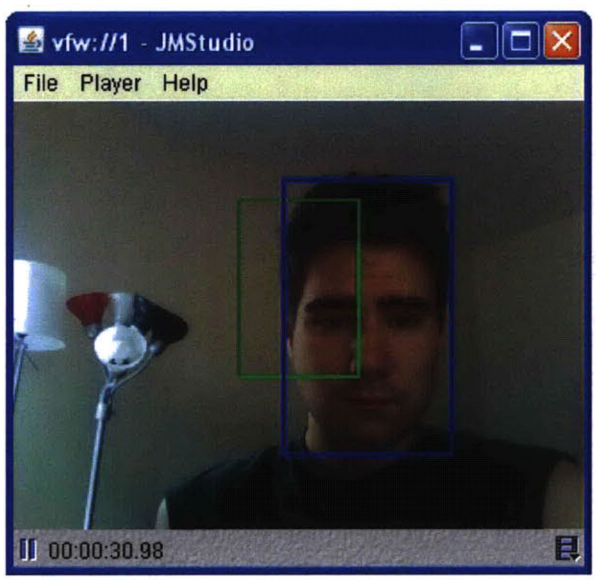

Figure 2-46: A screenshot of the video capture window with the face-tracking enabled. Here we can see the instantaneous face detection in green and the stable and filtered face detection in blue. The image within the blue rectangle was sent to the OQO device.

A solution to this problem was to build a persistency filter. A concept of a face was created along with variables for location, width and height. A collection of previously detected faces was maintained. Whenever a new detection was made, a search was performed to see if that face had been seen before, if so the face in the collection was adapted to the incoming parameters and its detection count increased. Every now and then, a purging would be performed where faces that had not been detected for a certain amount of time were removed. At any given time, the most relevant (most often or most recently detected) face was selected and its location and dimensions chosen for cropping the image and sending. A low-pass filter was used to smooth over the transition whenever a change was made in the selection of the most persistent face, filters were also employed to merge a detected face with its representation in the collection. This system worked very well for the purposes of our project. In figure 2-46 a sample video window can be seen where the most recent detection is painted in green and the most persistent one is painted in blue. The detection was only performed for every fifth frame because of computational efficiency reasons.

2.5.2 Custom Camera-Embedded Display

Eye-Contact

A fatal flaw of many current videoconferencing systems and telepresence robots is the mismanagement of eye-contact. Anybody that has used a videoconferencing tool recognizes the effect of having a discussion with somebody who doesn't look you in the eye. This can feel very disruptive. The effect is caused by the fact that while your partner is looking at your video stream on their computer screen, you are viewing theirs on your screen but that video is obtained from a camera that looks at them at an angle. It was our belief that by simply fixing the eye-contact problem with our system, we could improve the quality of telepresence interactions drastically.

Solution

A couple of solutions to this problem were considered. One option was do design a head-mounted camera. The benefits of this solution would be that the view of the operator's face would always be centered and viewed from a straight-on perspective even when they move their head or even shake or nod. The downside of this design is that it would require the operator to wear a head-mounted contraption which makes the system more intrusive and uncomfortable to operate.

Another option that was pursued, implemented and tested was to rather than render the operator's face on the robots head, render a 3D computer avatar that is modeled after the operator. The model would be animated by the movement and facial expressions of the operator. The benefits of this solution would be that now the system could have full control of how the operator is projected, that means that for example the operators could choose to have their avatar automatically animated if they choose not to control it themselves. Also the eye-gaze could now be faked in a way that it would appear that the operator is maintaining eye-contact with its remote partners.

Another benefit would be that head movement of the operator that can not be mapped to head movement of the robot could now be mapped onto the avatar while movement that does control the robot would not. Using real video prohibits this as all movement of the operator is revealed in the video even if some of it is being used to control the robot. This would avoid effects such as double nodding or double shaking of the head, that is when the operator nods or shakes their head, the remote partners observe the robot nodding or shaking but they also see in the video rendering the operator nodding or shaking. The downsides of this solution are for example that it takes a bit of time (10-15 minutes) to generate the 3D model of the operator and this would have to happen for every new operator. Also 3D models don't look as convincing as real video and lastly, rendering the 3D graphics on the OQO device is

Figure 2-47: A picture that shows a front view of the screen.

very computationally expensive.

The solution that was chosen was to build a display with a camera embedded in the center of it. The remote video window would be projected onto the center of this display by a video-projector placed in front of it. When the operator is controlling the robot and talking to their partner while watching them in the video window, the camera is looking right back at them under an angle that is very close to 0°. This gives the effect that the operator is looking straight forward from the perspective of a person watching the robot and when the head of the robot is faced towards a local participant, eye-contact can be established.

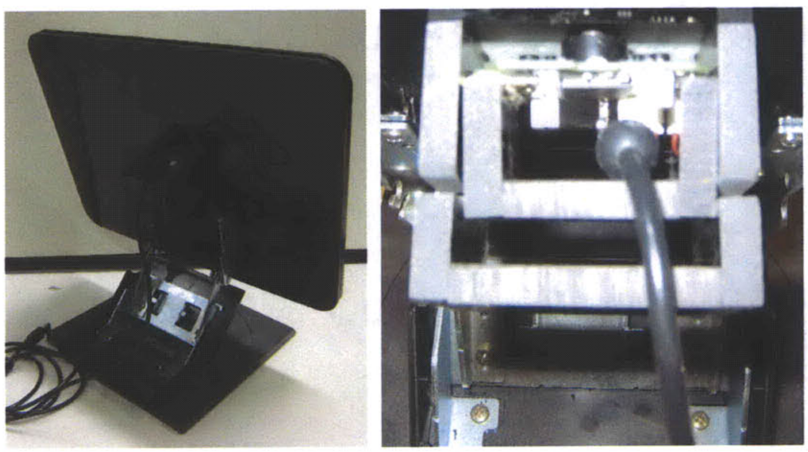

Figure 2-48: On the left a back view of the screen, on the right a close-up look at the camera and camera-mount.

Chapter 3

Evaluation

3.1 Introduction

Initially, our intent was to evaluate the fully working MeBot system against commercial systems that exist for similar purposes. Since, to the author's knowledge, there are no socially embodied telepresence robots commercially available, the systems we had intended to compare against were videoconferencing (*Skype*) and virtual worlds (*Second Life*). *Skype* was chosen because of its easy to use interface and widespread use. *Second Life* offer multiple ways of embodying people (virtually of course) and bringing them together in a way that can feel real even if they are physically separated by great distances.

As the development of the robot progressed, we realized that the system would probably not be in a state where a comparison like the one described above would be appropriate by the time of the thesis evaluation. A lot of time was spent developing the support systems needed to provide socially embodied presence (such as real-time audio/video transmission, motor control etc.) and during the development we realized that we wanted an evaluation that would support the further development of the robot rather than compare its absolute and final state against other mature platforms. In other words, we wanted to learn more about the system by evaluating it on its own before a comparison against other systems would be made.

There were multiple options to choose from in terms of which aspects of the system to evaluate. We could have tried evaluating different modes of operator control of the robot, different levels of autonomy in behavior of the robot or the usefulness of the robot for achieving different types of tasks remotely in collaboration with other people, just to list a few examples. The scenario that we chose to evaluate was one that would tell us something about the usefulness of the socially expressive movement of the robot. We decided to run a between-subjects study where participants would have an interaction with a person mediated through the robot. The participants would be collaborators of the robot-mediated person as opposed to operators of the robot. The two conditions of the study would have to do with the expressiveness of the robot. Basically in both conditions, the robot would be mobile on a table but in one condition the arms and head of the robot would be in a static pose where as in the other condition they would be moving corresponding to the operator's movements. The details of the experiment will be further explained chapters 3.3.2, 3.3.3 and 3.3.4.

When it had been decided which aspects of the robot to evaluate it was now important figure out which variables to measure and how. Which variables to measure was a question we thought hard about and a discussion is provided in section 3.2. How to measure these things is a different question all together and involves choosing a scenario to place the interaction in as well as a method to collect data. The situation needed to be a scenario where people were motivated to have an active conversation with the robot-mediated person, possibly make use of physical objects in the environment, make use of the mobility of the robot and preferably have some measure of success or other type of task-related measure that could be used to compare the experimental conditions.

We thought about several different situations including a task where the robot would either give or ask for directions using a map at the remote location or a block building task where the local collaborator and the robot operator would have to coordinate on building a specific structure that conforms to some "secrets" that the different collaborators would know and the robot operator had to find out. We ended up choosing a variation on the *desert survival task* for the evaluation. The task was originally designed by Lafferty, Eady and Elmers [Lafferty & Elmers, 1974] but the variation that we chose to use was designed by Takayama et al. [Takayama *et al.*, 2009].

3.2 Relevant Measures

3.2.1 Social Presence

One of the main goals of the presented work is to provide a more embodied medium for communication. That is to give people a way to be present in a space that is separated from them by distance, in a more immersive way than current technologies allow. Social presence has frequently been used as a quality measure of different communication media ([Hauber *et al.*, 2005], [Bente *et al.*, 2004]) and it is particularly relevant to our system for what we want to facilitate is exactly social expression and behavior.

3.2.2 Trust

An important application for telepresence are business meetings, this is the most common application of telepresence systems and has been the major focus for commercialization of products in this domain. With modern globalization of industries and governments, business is conducted between distant regions very frequently and in such transactions, trust is vital. This is why we thought that the measure of trust would be a useful metric to assess the success of our system.

3.2.3 Cooperation

Another application of telepresence that hasn't received quite the same amount of attention as business meetings but is equally or even more relevant to this technology are collaborative meetings. By collaborative we mean a more active meeting, such as that of designers, artists or developers. These types of meetings usually involve the active development of either a tangible product or idea that requires the complete engagement and participation of every collaborator. Cooperation is a measure that could be vital to the assessment of our system's compliance to the requirements of these types of situations.

3.2.4 Engagement

Throughout the design and idealization of this project we have had family communication in mind as an important application that needs significant attention. We want our system to facilitate family communication that allows a deeper engagement and a more enjoyable experience than that which can be obtained via simple videoconferencing. Engagement is a measure that is relevant in this sense.

3.3 Method

3.3.1 Participants

A total of 48 people participated in the study, six participants' data was excluded from the analysis leaving a total of $n = 42$ participants. Exclusion criteria was set by two rules: If participant experienced severe technical difficulty with the robot or if it was found out that they knew the operator from before the time of the study, they would be excluded.

Participants were asked to answer questions that provided demographic information, the resulting statistics are presented in table 3.1. Participants were asked to rank their knowledge about robotics on a seven point scale ($\overline{M} = 3.80$, $\sigma = 1.52$). They were also asked to rank their knowledge about computers on a seven point scale ($\overline{M} = 5.18$, $\sigma = 1.36$). The participants were asked for their age in years ($\overline{M} = 23.21$, $\sigma = 8.92$).

3.3.2 Task

The desert survival problem was originally developed by Lafferty and Eady [Lafferty & Elmers, 1974] but has since been used by several social scientists and roboticists ([Takayama *et al.*, 2009], [Bente *et al.*, 2004], [Biocca *et al.*, 2001], [Hauber *et al.*, 2005] and [Kidd & Breazeal, 2004]). We slightly modified a version of the task designed by Takayama et al. [Takayama *et al.*, 2009].

The back story that was presented to participants was that they and their partner (the robot operator) had been traveling on an airplane when the airplane started

Table 3.1: Study participants statistics

Group	n	Percentage
Gender		
Female	24	57.14%
Male	18	42.86%
Ethnicity		
Asian	13	30.95%
African American	5	11.90%
Hispanic	3	7.14%
Caucasian	16	38.10%
Other	5	11.90%
Level of education		
Some high school	1	2.38%
High school degree	1	2.38%
Some college	22	52.38%
College degree	8	19.05%
Some graduate school	6	14.29%
Graduate school degree	4	9.52%
Use of videoconferencing		
Never	14	33.33%
Less than once a month	9	21.43%
1-4 times a month	8	19.05%
5-10 times a month	8	19.05%
11-20 times a month	3	7.14%

experiencing problems which resulted in a crash in the middle of a desert. The only location information they had is that the nearest town was about seventy miles away and that they had a rough heading for the town. They were both uninjured but they were the only survivors of the crash and there was nothing useful that could be salvaged from the wreck of the airplane.

The participants were then asked to think about which items they would like to have packed in their bag for this trip knowing that they would have to survive in the desert. The table in front of the participants had twelve items neatly arranged in six pairs as is shown in figure 3-2, participants were asked to look at each pair individually and choose which object out of the two they would rather bring with them. Before discussing their choices with their partner, they would fill out an initial choice list (the list can be viewed in appendix A). After filling out the list, they would discuss their choices with the robot operator which would provide advise on the object selection according to a pre-made script. Finally the participant was asked to mark their final choice of items on a final choice list, this was where they would make changes to their initial choice if they cared to do so or stick with the initial choice.

3.3.3 Setup

The study was a doubly blind study, that is the operator was not aware of the hypotheses nor the conditions in any of the experiments. For consistency, the robot was always operated by the same person, an undergraduate researcher in the lab Nancy Foen. The operator controlled the robot through all of the interactions from a private office in our lab, the office setup can be seen in figure 3-1. The office door was closed so that participants wouldn't see the operator in person until after the experiment. The operator used three interfaces to control the robot, a sympathetic controller for the shoulders and arms of the robot to control those DOFs, head-pose and orientation capture for controlling the neck and head and a graphical interface to control those DOFs in case either of the other two would stop functioning, the interfaces are explained in chapter 2.3.

Much emphasis was placed on the operator performing consistently between interactions and especially between conditions. Effort was made to make the operator unaware of which condition was being performed. The camera on the robot was moved from the head which is where it would normally be, down to the base so that the operator wouldn't notice the camera view change as she moved the head of the robot. Noise was added to the operators audio signal so that she would not hear the motor noise that resulted from her moving the arms or head of the robot. A video noise filter (to add noise, not filter it out) was designed to make it harder to detect

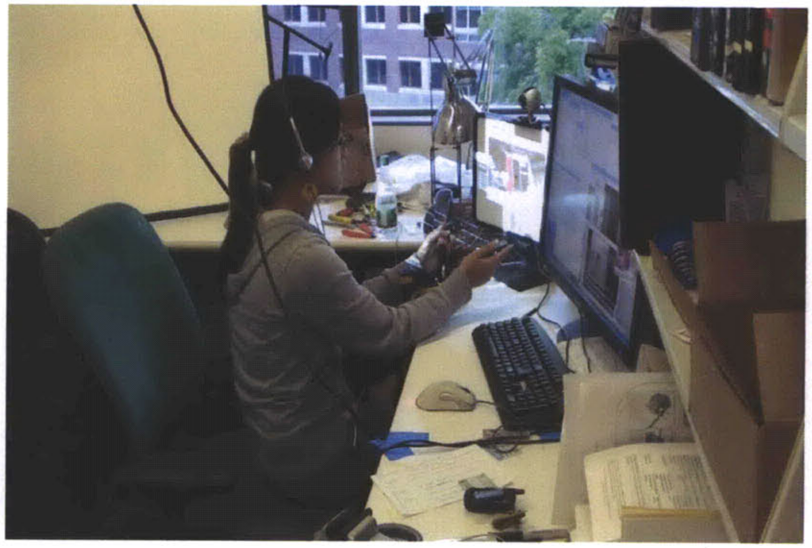

Figure 3-1: A picture of the operator Nancy Foen and the setup in the operator station. Nancy has the control interface in front of her, as well as the space navigator for mobility and the sympathetic for expression. She also has the remote scene projected on the custom camera-embedded screen.

motion in the camera view when the motors moved abruptly, it was found that the video didn't really show jitter when the motors moved so it could be rendered without the noise filter. The operator was videotaped so that her behavior could be monitored for consistency between conditions.

Figure 3-2: A picture of the robot station. Participants would sit in the red chair and have a conversation with Nancy through the robot (to the left in figure). Items are arranged in pairs in front of the participant.

The participants were seated in a space that had been partitioned from the rest of the lab by black curtains. They would be seated at the center of the long side of an approximately six foot long table. Across from the participant, on the other side of the table was a smaller table on which the robot was placed. Along the long table were twelve items arranged neatly in six pairs of two items, the setup can be viewed in figure 3-2. The pairs contained the following items:

- a tarp and a canvas
- a knife and a gun
- a chocolate bar and a water bottle
- a compass and a mirror
- a flashlight and matches
- a map and a book on edible animals

The operator performed as closely as she possibly could in accordance with a script for every interaction. The only reason for her to digress from the script would be to respond to unanticipated questions so that the participant experienced natural dialog. The script contained arguments for and against every item that was available for the participants to select. The operator would always ask the participant which object out of every pair they chose before disclosing which item she had chosen. She would always disagree with the participant on the first two items and give arguments for bringing the item that the participant did not choose. On the third pair she would agree, disagree on the next two items again and finally agree on the last one. This way, she would have disagreed with every participant on the same four pairs and we could then investigate how many participants would decide to change their initial selections on these four items after having heard the scripted arguments for the other items.

3.3.4 Protocol

Recruitment for the study was mostly conducted by sending advertisements to mailing lists around the Media Lab as well as several of the undergraduate dorms and some summer programs at MIT. Flyers were also posted at various locations around campus like the student center, main hallways and dorms. A registration website was open during recruitment for managing available timeslots and contact information for possible participants.

In a follow-up email to a participant's on-line registration, directions were given on how to get to the Personal Robots' lab in the Media Lab. When a participant would walk through the door of the lab at the time of their appointment, they would be greeted by the author and given a consent form. They were asked to read the form carefully and sign their consent at the last page if they chose to do so. The author would step away to give them privacy while they would read the consent form.

While the participants were reading the form, the author would go into the operator's office and flip a coin to determine the condition of this particular run of the study. When the operator would step out of the office, the chosen condition would be asserted. The author would then turn on the operator's camera and tell the operator that she was now being recorded. The author would then step out and turn on the camera that was facing the robot and the study participant as well as making sure that the robot worked, especially the audio/video connection with operator.

When the participant was done signing the consent form, the author would ask them to follow him into the partitioned-off space. They would then be asked to take a seat and the author would introduce them to the operator, Nancy. They were told that Nancy was going to be their partner in the task of the study but that she couldn't attend in person and was therefore joining us via the MeBot platform. The author would then recite the following script:

You and your partner Nancy are traveling on an airplane when the pilot announces that the airplane is experiencing some problems. Soon after, the airplane crashes and you find yourself stranded in the middle of the desert. There are no survivors except for you and Nancy but miraculously, neither of you were injured in the crash. Nothing useful can be salvaged from the wreck of the airplane so the only items available to you are the ones you packed in your backpack. The only location information you have is that there is a town about seventy miles off your current position and you have a rough heading toward the town.

You are now faced with the task of surviving in the desert and you can choose to stay put and try to get rescued, try to walk to the town or any other strategy of your liking. I will now ask you to think about which items you would like to bring with you on this trip, knowing that you will be faced with the challenge of surviving in the desert.

I have here twelve items that could be useful in this type of situation, the items are arranged in six pairs of two items. Please consider every pair on its own and determine which out of the two items you would rather bring with you on this trip. Please make your initial selections on this form [present the initial-selection form] before discussing your choices with Nancy, after you have filled the form out let Nancy know and you guys will discuss your choices and hers. If you ever wish to revise any of your initial options, please do not change your form because once you and Nancy have discussed all of the pairs you will be presented with a final-selection form where you can mark your final selection and make any changes you wish.

So just let Nancy know when you are finished and good luck !

After reciting the script, the author would exit the partitioned space but stay close to monitor the interaction. If any technical difficulties would arise, the author would be able to tend to them immediately as well as know when to enter and present the final-selection form.

The author would step away to give the participant space to finish the final-selection form and then he would tell the participant that now the first part of the study was over and we would be leaving the partitioned space. They would then say goodbye to Nancy and walk in to the general lab space. The author would thank the participant for completing the first part of the study and tell them that the second part was a questionnaire that would be used to analyze how they had experienced the interaction (the questionnaire can be viewed in appendix A). The author would explain the format of the questions and tell them that everything was completely confidential, they were also asked not to spend too much time on any one question but rather give their first impression on them as there were many questions. They would be asked to read the instructions carefully and when they were done they should just signal any of the researchers. At that point they were asked to sign a receipt for $10 which they would then receive for their participation.

If the participants had experienced the passive condition of the robot, they would always be de-briefed where the point of the research was explained to them and the two conditions were exposed so they would also experience the robot in its fully working mode. The people that experienced the fully expressive condition would also be de-briefed but they didn't need as much of an explanation as they had already experienced the robot in a fully functioning mode.

3.3.5 Dependent Measures

The questionnaire that was presented to participants after the interaction with the robot had seven distinct question groups, three of which had subgroups of questions. About half of the questions originated from a fairly widely accepted measure of social presence called *The Networked Minds Measure of Social Presence* [Biocca *et al.*, 2001] by Biocca, Harms and Gregg. Fifteen questions were used to measure trust, they came from Wheeless's and Grotz's *Individualized Trust Scale* as reported in [Rubin *et al.*, 2004]. General engagement was measured by five questions that were taken from the six aspects of presence in [Lombard *et al.*, 2000]. Cooperation was measured by three questions that were obtained from a questionnaire used in [Takayama *et al.*, 2009], other ungrouped questions were also obtained from that study. The only dependent measure that was particular to the desert survival task was *number of changed items*, as was explained in previous sections participants made an initial selection of items and later a final selection after having listened to their partner's reasoning. The number of items that they changed from their initial to final selection was measured as a dependent variable.

3.4 Results

The dependent variables

The results of the evaluation are presented in this section. As was explained in section 3.3.5 the dependent measures are: Co-presence, psychological involvement, behavioral engagement, trust, general engagement, cooperation and the task specific measure of number of items changed. The questionnaire is presented in the appendix.

How to read these results

The questions were answered on a a seven point Likert scale. The only processing performed on any of the data is that some of the scales need to be reversed before entered into averages as those particular questions require a low score to signal a high level of presence, engagement or whatever the measure is.

Example: If the question"Did you feel deeply engaged in the interaction" would report a score of 1, it would mean that people didn't feel very engaged. If the question "Did you feel very distant from your partner" would score a 1, it would mean that people actually felt that they were distant from their partner because the score for this question would be inverted because of how it is presented.

Method of analysis

The results were analyzed using a method called *Analysis of Variance* (ANOVA), a fairly well known method to determine if samples from two groups actually originate from the same population (null-hypothesis) or if there is a statistically significant difference in their means. One of the outcomes of this analysis is the p measure which tells us the likelihood of this particular outcome of differences in the group means. If p is close to 1 then there is high likelihood that this difference would show up at random, if $p < 0.05$ then there is less than 5% probability that this result could be caused by chance. For all measures the $n_{expressive} = 23$ and $n_{static} = 19$. The difference in number of subjects in each condition is because unfortunately, all participants who experienced technical difficulty or a disturbance that forced their data to be excluded from the analysis experienced the static condition.

Conditions

The two conditions being compared were - a mobile robot that is static in pose, it doesn't move arms or neck (referred to as OFF in tables) and a mobile robot that is expressive, it the moves arms and neck in accordance to Nancy's commands (referred to as ON in tables).

3.4.1 Co-Presence

It is clear from graph 3-3 that there is no difference in the means of the rankings from the two groups in the measure of co-presence. This is confirmed by the ANOVA results in table 3.2. None of the questions in table 3.3 indicated a significant difference on their own when analyzed using the $t-test$ and neither did the subgroups of questions for this measure.

Table 3.2: ANOVA results for the measure of co-presence

Question group	p
Isolation/aloneness	0.133
Mutual awareness	0.143
Attentional allocation	0.711
Co-Presence Average	**0.164**

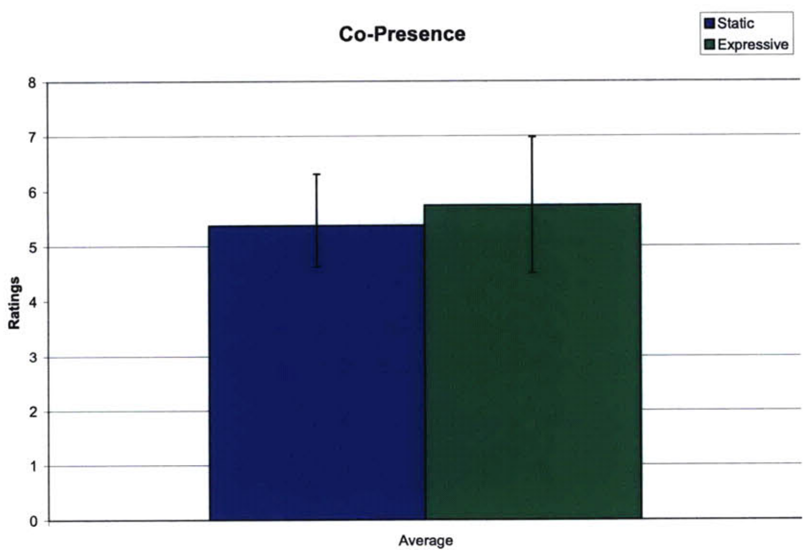

Figure 3-3: A graph showing the average rating of question groups measuring co-presence. Error bars indicate $\pm\sigma$. $p < 0.165$.

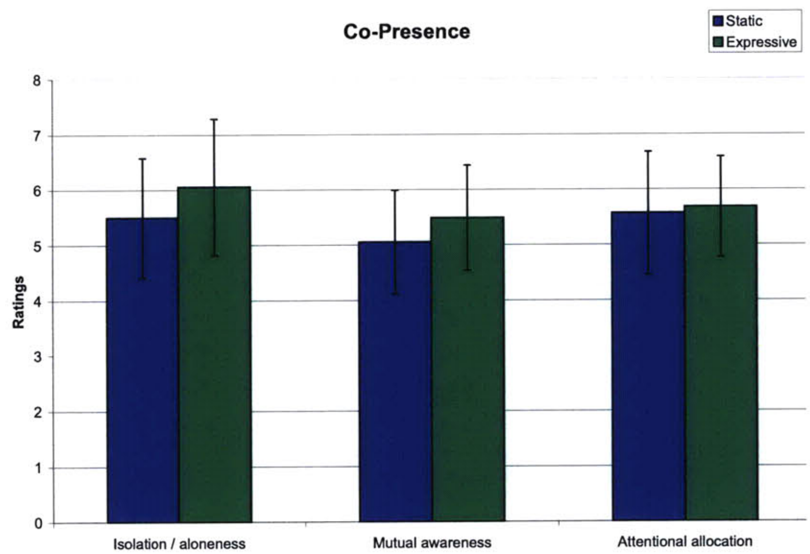

Figure 3-4: A graph showing the rating for individual question groups measuring co-presence. Error bars indicate $\pm\sigma$.

Table 3.3: Questions to measure co-presence, means and standard deviations for both conditions

No.	Question	Off $\overline{M}$	Off σ	On $\overline{M}$	On σ
	Isolation/aloneness				
1.	I often felt as if I was all alone	5.684	1.376	6.196	1.286
2.	I think the other individual often felt alone	5.316	1.057	5.913	1.276
	Group Average	**5.500**	**1.080**	**6.054**	**1.234**
	Mutual Awareness				
1.	I hardly noticed another individual	6.052	1.268	6.435	1.308
2.	The other individual didn't notice me	6.158	1.015	6.130	1.392
3.	The other individual didn't notice me	3.316	1.734	4.261	2.158
4.	I was often aware of others in the environment	3.789	1.653	4.000	2.000
5.	Others were often aware of me in the room	5.684	1.376	6.196	1.286
6.	I often felt as if I was all alone	5.684	1.376	6.196	1.286
7.	I think the other individual often felt alone	5.316	1.057	5.913	1.276
	Group Average	**5.053**	**0.933**	**5.489**	**0.950**
	Attentional Allocation				
1.	I sometimes pretended to pay attention to the other individual	4.842	1.979	5.152	1.927
2.	The other individual sometimes pretended to pay attention to me	4.947	1.615	5.152	1.675
3.	The other individual paid close attention to me	5.684	0.946	5.804	1.467
4.	I paid close attention to the other individual	5.842	1.119	5.891	1.381
5.	My partner was easily distracted when other things were going on around us	5.842	1.463	5.783	1.043
6.	I was easily distracted when other things were going on around me	5.263	1.821	5.130	1.740
7.	The other individual tended to ignore me	6.158	1.119	6.391	0.656
8.	I tended to ignore the other individual	5.947	1.508	6.152	1.327
	Group Average	**5.566**	**1.112**	**5.682**	**0.908**
	Co-Presence Average	**5.373**	**0.936**	**5.742**	**0.749**

3.4.2 Psychological Involvement

For the dependent measure of psychological involvement, a statistically significant difference was found in the mean rating for participants from the two conditions. This can be observed in figure 3-5 and is confirmed in table 3.4.

Table 3.4: ANOVA results for the measure of psychological involvement

Question group	p
Empathy	0.103
Mutual understanding	0.033
Psychological Involvement Average	**0.022**

The subgroup for mutual understanding shows a significant difference ($p < 0.034$) while the subgroup for empathy does not ($p < 0.104$). However, two questions in that subgroup do show a significant difference: question no. 3 a particularly strong difference ($p < 0.002$) and question no. 6 also ($p < 0.057$).

All differences in measures of psychological involvement are in favor of the expressive condition of the study.

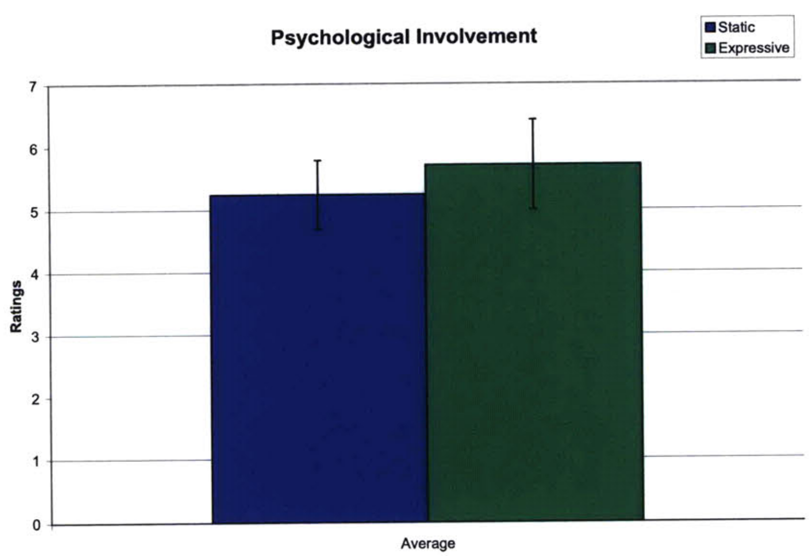

Figure 3-5: A graph showing the average rating of question groups measuring psychological involvement. Error bars indicate $\pm\sigma$. $p < 0.023$.

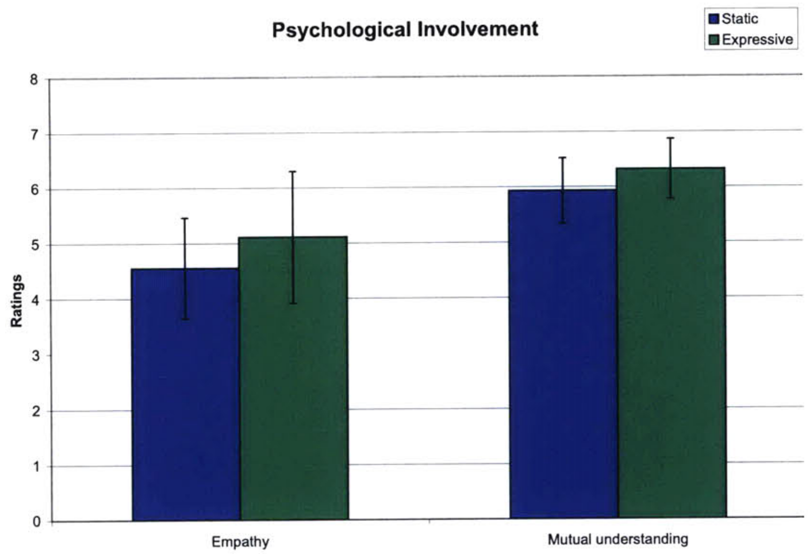

Figure 3-6: A graph showing the rating for individual question groups measuring psychological involvement. Error bars indicate $\pm\sigma$.

Table 3.5: Questions to measure psychological involvement, means and standard deviations for both conditions

No.	Question	Off $\overline{M}$	Off σ	On $\overline{M}$	On σ
	Empathy				
1.	When I was happy, the other was happy	4.842	1.344	5.478	1.082
2.	When the other was happy, I was happy	5.053	1.177	5.435	1.080
3.	The other individual was influenced by my moods	4.000	1.054	5.217	1.242
4.	I was influenced by my partner's moods	4.947	1.026	5.043	1.581
5.	The other's mood did NOT affect my mood	4.526	1.712	4.630	1.720
6.	My mood did NOT affect the other's mood	3.947	1.393	4.848	1.548
	Group Average	**4.552**	**0.910**	**5.109**	**1.193**
	Mutual Awareness				
1.	My opinions were clear to the other	5.684	1.057	6.217	0.795
2.	The opinions of the other were clear	6.263	0.733	6.217	0.795
3.	My thoughts were clear to my partner	5.526	1.349	6.283	0.580
4.	The other individual's thoughts were clear to me	5.526	1.504	6.326	0.633
5.	The other understood what I meant	6.158	0.602	6.391	0.583
6.	I understood what the other meant	6.421	0.607	6.478	0.511
	Group Average	**5.241**	**0.548**	**5.714**	**0.711**
	Psychological Involvement Average	**5.241**	**0.548**	**5.714**	**0.711**

3.4.3 Behavioral Engagment

The measure of behavioral engagement showed a significant difference as can be seen in figure 3-7 and is confirmed in table 3.6.

Table 3.6: ANOVA results for the measure of behavioral engagment

Question group	p
Behavioral interdependence	0.495
Mutual assistance	0.042
Dependent action	0.286
Behavioral Engagement Average	**0.024**

We can also see that the difference between the means of the subgroups of this measure is not as strongly significant as the average which is an interesting result and shows that the accumulation of the data in these subgroups adds to reduce the variance in either condition which fortifies our belief that there underlying populations are truly different.

All differences in the measures of behavioral engagement are in favor of the expressive condition of the study.

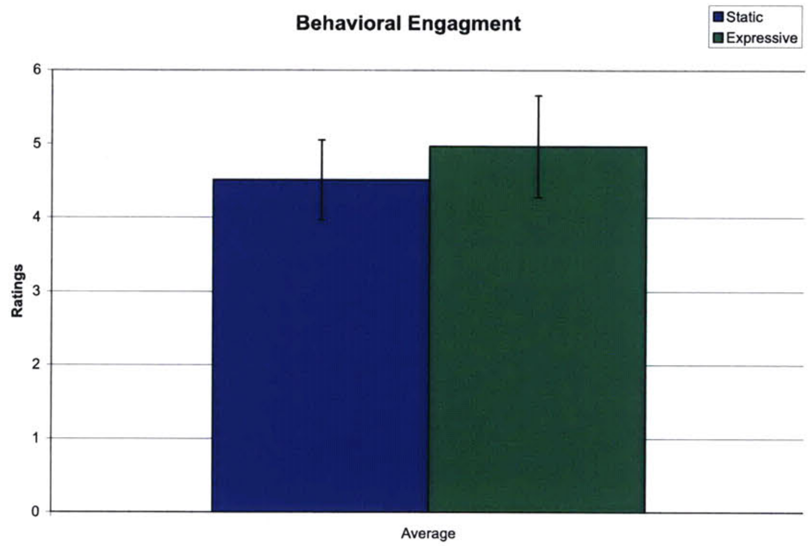

Figure 3-7: A graph showing the average rating of question groups measuring behavioral engagment. Error bars indicate $\pm\sigma$. $p < 0.024$.

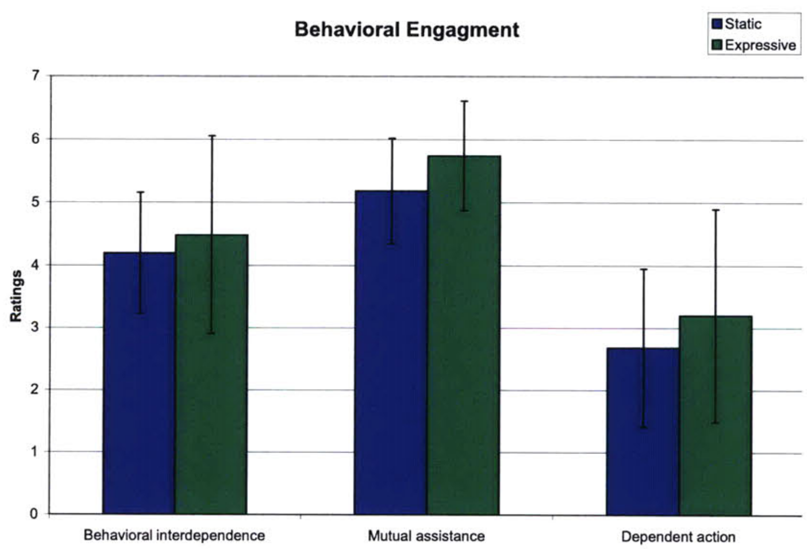

Figure 3-8: A graph showing the rating for individual question groups measuring behavioral engagement. Error bars indicate $\pm\sigma$.

Table 3.7: Questions to measure behavioral engagement, means and standard deviations for both conditions

No.	Question	Off $\overline{M}$	Off σ	On $\overline{M}$	On σ
	Behavioral interdependence				
1.	My actions were dependent on the other's actions	3.947	1.471	4.370	1.625
2.	The other's actions were dependent on my actions	3.368	1.383	4.174	1.723
3.	My behavior was in direct response to the other's behavior	4.579	1.387	4.543	1.712
4.	The behavior of the other was I direct response to my behavior	4.053	1.268	4.457	1.815
5.	What the other did affected what I did	5.105	1.197	4.761	1.783
6.	What I did affected what the other did	4.105	1.487	4.565	1.727
	Group Average	**4.193**	**0.964**	**4.478**	**1.577**
	Mutual assistance				
1.	My partner did not help me very much	5.316	1.204	5.630	1.316
2.	I did not help the other very much	3.947	1.268	4.674	1.345
3.	My partner worked with me to complete the task	5.684	1.057	6.348	1.027
4.	I worked with the other individual to complete the task	5.789	0.976	6.304	0.876
	Group Average	**5.184**	**0.837**	**5.739**	**0.868**
	Dependent action				
1.	The other could not act without me	2.632	1.571	3.630	2.186
2.	I could not act with the other	2.737	1.593	2.761	1.705
	Group Average	**2.684**	**1.272**	**3.196**	**1.704**
	Behavioral Engagement Average	**4.509**	**0.544**	**4.968**	**0.694**

3.4.4 Trust

No statistically significant difference was detected for the measure of trust as can be seen in figure 3-9 and table 3.8. We will go into discussion as to possible reasons for this in section 3.5.

Table 3.8: ANOVA results for the measure of trust

Question group	p
Trust	**0.959**

Further investigation into the individual effects of every question in this measure using the $t - test$ revealed that no question showed any significant difference.

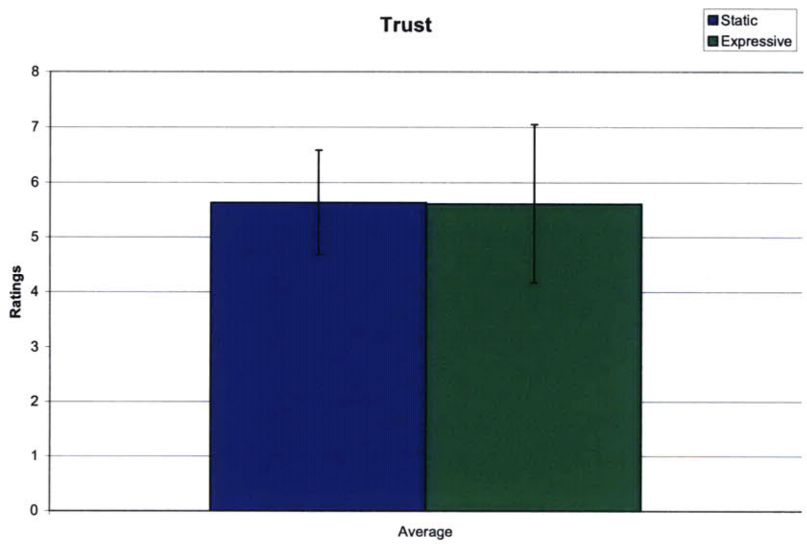

Figure 3-9: A graph showing the average rating of question groups measuring trust. Error bars indicate $\pm\sigma$. $p < 0.959$.

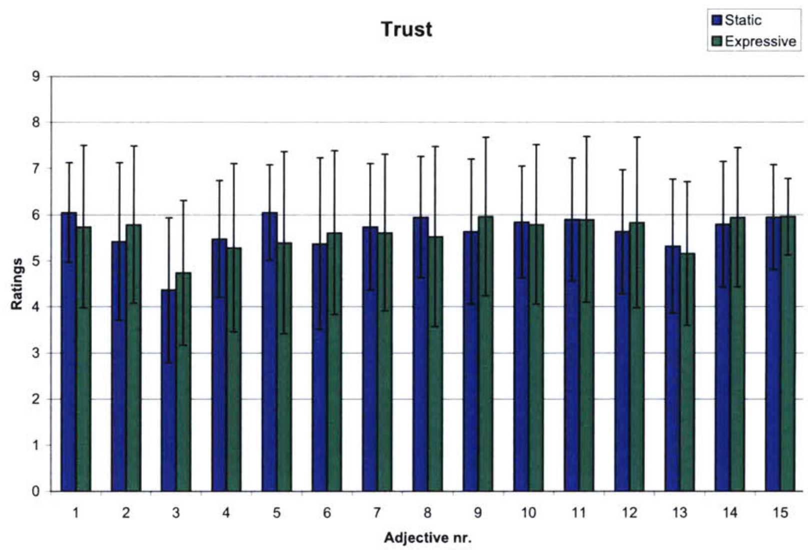

Figure 3-10: A graph showing the rating for individual question measuring trust. Error bars indicate $\pm\sigma$.

Table 3.9: Questions to measure trust, means and standard deviations for both conditions

No.	Adjective	Off $\overline{M}$	Off σ	On $\overline{M}$	On σ
1.	Trustworthy	6.053	1.079	5.739	1.764
2.	Trustful of this person	5.421	1.710	5.783	1.704
3.	Confidential	4.368	1.571	4.739	1.573
4.	Benevolent	5.474	1.264	5.283	1.827
5.	Safe	6.053	1.026	5.391	1.971
6.	Candid	5.368	1.862	5.609	1.777
7.	Not Deceitful	5.737	1.368	5.609	1.699
8.	Straightforward	5.947	1.311	5.522	1.951
9.	Respectful	5.632	1.571	5.957	1.718
10.	Considerate	5.842	1.214	5.783	1.731
11.	Honest	5.895	1.329	5.891	1.796
12.	Reliable	5.632	1.342	5.826	1.850
13.	Faithful	5.316	1.455	5.152	1.563
14.	Sincere	5.789	1.357	5.935	1.510
15.	Careful	5.947	1.129	5.957	0.825
	Trust Average	**5.632**	**0.950**	**5.612**	**1.437**

3.4.5 General Engagement

The dependent measure of general engagement showed a statistically significant difference in the means of ratings between the two conditions. This can be seen in figure 3-11 and is confirmed in table 3.10

Table 3.10: ANOVA results for the measure of general engagement

Question group	p
General Engagement	**0.015**

The questions that made the biggest contribution to the significant difference between the groups were questions 1, 3, and 5.

All differences in the measures of general engagement are in favor of the expressive condition of the study.

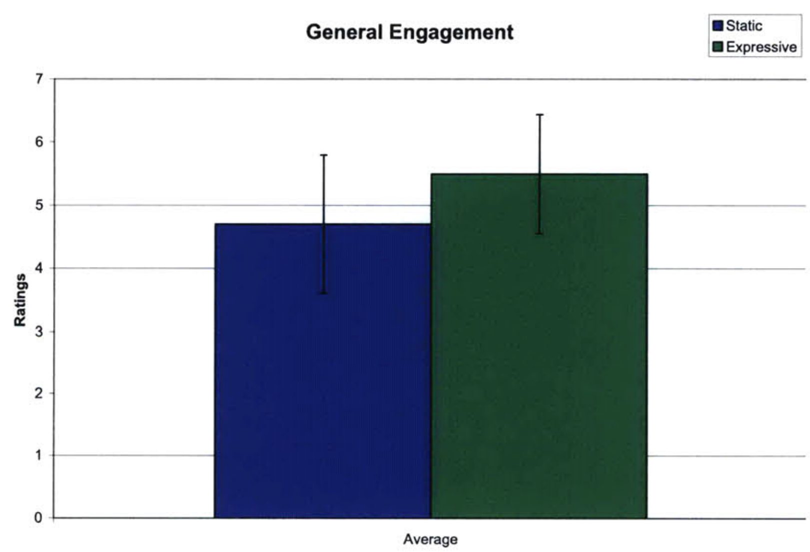

Figure 3-11: A graph showing the average rating of question groups measuring general engagement. Error bars indicate $\pm\sigma$. $p < 0.016$.

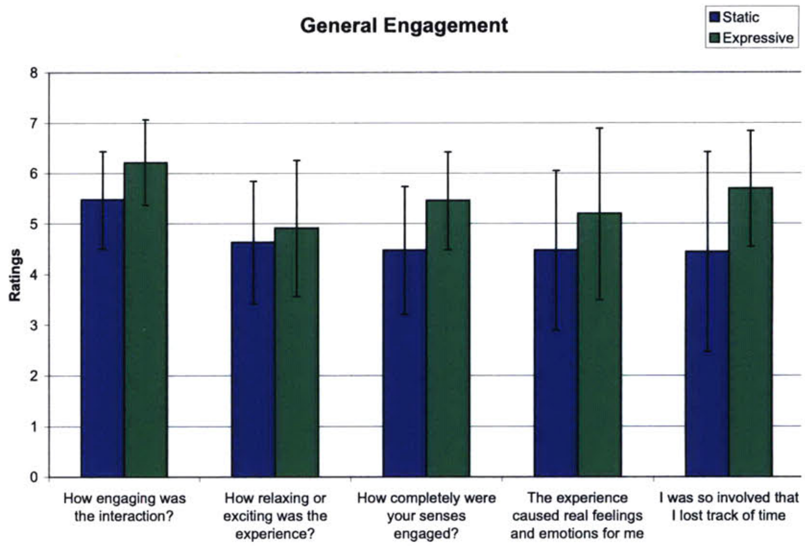

Figure 3-12: A graph showing the rating for individual question measuring general engagement. Error bars indicate $\pm\sigma$.

Table 3.11: Questions to measure general engagement, means and standard deviations for both conditions

No.	Question	Off $\overline{M}$	Off σ	On $\overline{M}$	On σ
1.	How engaging was the interaction	5.474	0.964	6.217	0.850
2.	How relaxing or exciting was the experience	4.632	1.212	4.913	1.345
3.	How completely were your senses engaged	4.474	1.264	5.457	0.964
4.	The experience caused real feelings and emotions for me	4.474	1.577	5.196	1.697
5.	I was so involved that I lost track of time	4.447	1.978	5.696	1.146
	General Engagement Average	**4.700**	**1.089**	**5.496**	**0.944**

3.4.6 Cooperation

The dependent measure of cooperation had the fewest questions in its group but it showed the strongest difference in means between conditions. ANOVA results report a strong statistical significance. This can be observed in figure 3-13 and is confirmed in table 3.12.

Table 3.12: ANOVA results for the measure of cooperation

Question group	p
Cooperation	**0.005**

The question that made the least contribution to the significant difference between the groups was question no. 3, which asks for how much the other person cooperated with the participant.

All differences in the measures of Cooperation are in favor of the expressive condition of the study.

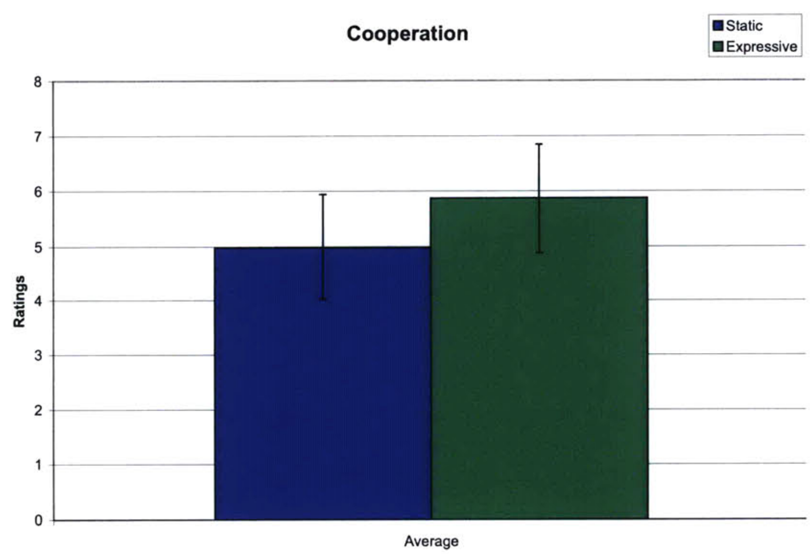

Figure 3-13: A graph showing the average rating of question groups measuring general cooperation. Error bars indicate $\pm\sigma$. $p < 0.006$.

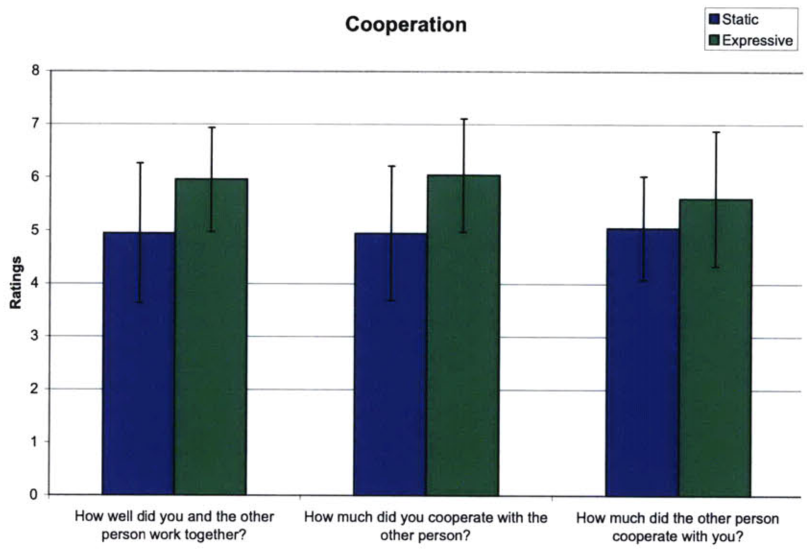

Figure 3-14: A graph showing the rating for individual question measuring general cooperation. Error bars indicate $\pm\sigma$.

Table 3.13: Questions to measure Cooperation, means and standard deviations for both conditions

No.	Question	Off $\overline{M}$	Off σ	On $\overline{M}$	On σ
1.	How well did you and the other person work together	4.947	1.311	5.957	0.976
2.	How much did you cooperate with the other person	4.947	1.268	6.043	1.065
3.	How much did the other person cooperate with you	5.053	0.970	5.609	1.270
	Cooperation Average	**4.982**	**0.959**	**5.870**	**0.978**

3.4.7 Ungrouped Questions

In this section the results of individual questions is shown. These questions could not be placed in groups of other questions measuring the same thing which would allow for averaging and obtaining a more general result but that is not to say that they don't show meaningful results.

Table 3.14: ANOVA results for the ungrouped questions

No.	Question group	p
1.	I really wanted to reach agreement with the other person	0.339
2.	How competitive was the other person	0.035
3.	How competitive were you	0.043
4.	How similar were the other person's suggestions to your own suggestions	0.020
5.	How comfortable did you feel during this activity	0.020
6.	How comfortable did the other person seem to be feeling during this activity	0.082
7.	How enjoyable was your experience with the MeBot system	0.008

Table 3.14 shows a statistically significant difference in almost all questions except for no. 1 and possibly no. 6 although that one is reasonably close to showing a significant difference. From figure 3-15 it can be seen that the only questions that show difference that is not in favor of the expressive case (excluding question no. 1) are the ones that ask for competitiveness. This will be further discussed in section 3.5.

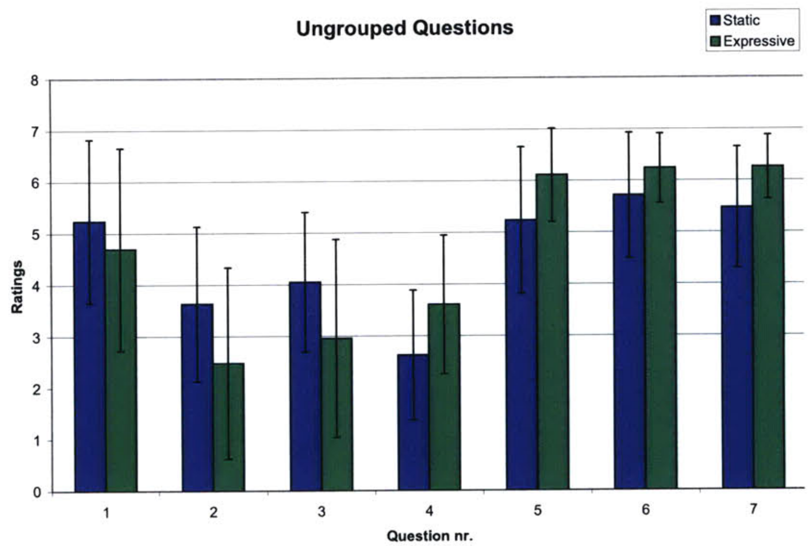

Figure 3-15: A graph showing the rating for individual ungrouped questions. Error bars indicate $\pm\sigma$.

Table 3.15: Ungrouped questions, means and standard deviations for both conditions

No.	Question	Off $\overline{M}$	Off σ	On $\overline{M}$	On σ
1.	I really wanted to reach agreement with the other person	5.237	1.584	4.696	1.964
2.	How competitive was the other person	3.632	1.499	2.478	1.855
3.	How competitive were you	3.947	1.353	5.043	1.918
4.	How similar were the other person's suggestions to your own suggestions	5.368	1.257	4.391	1.340
5.	How comfortable did you feel during this activity	5.237	1.418	6.109	0.904
6.	How comfortable did the other person seem to be feeling during this activity	5.711	1.217	6.239	0.672
7.	How enjoyable was your experience with the MeBot system	5.474	1.172	6.261	0.619

3.4.8 Number of Changed Items

The task-specific measure of number of changed items did not show a significant difference between conditions. This can be seen in figure 3-16 and is confirmed in table 3.16.

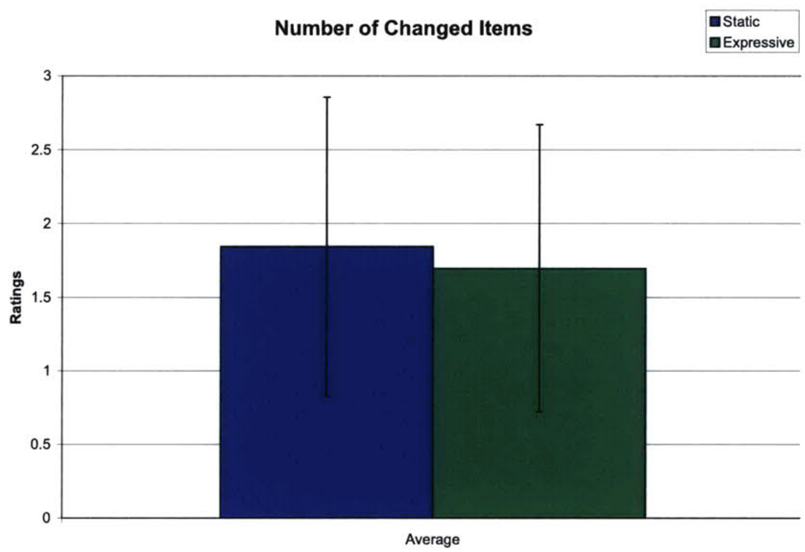

Figure 3-16: A graph showing the rating for number of changed items. Error bars indicate $\pm\sigma$. $p < 0.637$.

Table 3.16: ANOVA results for the ungrouped questions

Dependent measure	p
Number of Changed Items	0.637

3.5 Discussion

Before running this study, we set forth the following research hypotheses:

1. We believe that people will experience stronger co-presence using an expressive telerobot.
2. We believe that people will experience more psychological involvement with their partner when using an expressive telerobot.
3. We believe that people will feel more behaviorally engaged with their partner when using an expressive telerobot.
4. We believe that people will trust their partner more when using an expressive telerobot.
5. We believe that people will feel generally more engaged with their partner when using an expressive telerobot.
6. We believe that people will cooperate better with their partner when using an expressive telerobot.
7. We believe that people will enjoy their interaction more using an expressive telerobot.

3.5.1 Social Presence

The measure of social presence was quantified along the dimensions of co-presence, psychological involvement and behavioral engagement as in [Biocca *et al.*, 2001].

Co-Presence

"The degree to which the observer believes he/she is not alone and secluded, their level of peripherally or focally awareness of the other, and their sense of the degree to which the other is peripherally or focally aware of them" [Biocca *et al.*, 2001].

This dependent measure included three subgroups of questions: Isolation/aloneness, mutual awareness and attentional allocation. None of the subgroups showed a significant difference between the groups nor did the averaged measure of co-presence.

It is possible to imagine that the effect of the difference between an expressive and static robot is too minimal when compared with the effect of the embodiment of a robot. To elaborate, the author wondered if the fact that both conditions consisted of an embodied partner via a mobile robot that looks capable of expressive communication had a strong enough effect on the measure of presence that the difference between an expressive and static robot was too "fine grain" to detect a difference. This issue will be addressed in a future study with the same interaction but no robot only a device, further explained in section 4.2.

The author noticed that this portion of *the networked minds measure of social presence* questionnaire seemed to be better suited for virtual environments than for robot-mediated communication. Questions like "I hardly noticed another individual" and "I sometimes pretended to pay attention to the other individual" are not as valid for the robotic situation as they are for virtual environments. The author also noticed a lot of confusion with many participants when it came to answering the following two questions: "I was often aware of others in the environment" and "Others were often aware of me in the room", their meaning is not obvious in the context of a robot-mediated conversation and that might have affected the results.

Hypothesis number one was not upheld.

Psychological Involvement

"The degree to which the observer allocates focal attention to the other, empathically senses or responds to the emotional states of the other, and believes that he/she has insight into the intentions, motivation, and thoughts of the other." [Biocca *et al.*, 2001].

Psychological involvement was measured along two dimensions: Empathy and mutual understanding. Empathy didn't show a significant difference but one of the questions in that group did produce the strongest difference of all of the questions in the whole questionnaire, in favor of the expressive condition. The question was "The other individual was influenced by my moods" ($p < 0.002$). Mutual understanding showed a significant difference, as did the averaged measure of psychological involvement.

One of the research claims we made in the beginning was that non-verbal communication such as body language, gaze, socially expressive gesturing etc. could be considered media for information in a conversation. Allowing for these channels of communication should therefore increase the amount of information that gets passed between people engaged in conversation. It did therefore not surprise the author to see that participants that experienced the expressive condition reported higher values of understanding, clarity and emotional influence in the interaction.

Hypothesis number two was upheld.

Behavioral Engagement

"The degree to which the observer believes his/her actions are interdependent, connected to, or responsive to the other and the perceived responsiveness of the other to the observer's actions." [Biocca *et al.*, 2001].

Behavioral engagement included three subgroups of questions: Behavioral interdependence, mutual assistance and dependent action. Mutual assistance was the only subgroup to show a statistically significant difference as well as the resulting averaged measure of behavioral engagement.

The results suggest that a socially expressive robot elicits more help and cooperation from its partner as well as being perceived more helpful by the partner, this is an interesting finding as one could just as well have believed that a robot that seems less capable might elicit more helpful behavior from their partner.

Hypothesis number three was upheld.

3.5.2 Trust and Cooperation

As previously discussed, important applications of telepresence systems are business meeting and collaborative meetings of designers. Very relevant measures for these situations are trust and cooperation.

Trust was measured by presenting participants with fifteen sets of antonyms (ex. unreliable - reliable) and asking them to rate the operator as they experienced her through this system on a seven-point Likert scale. The result did not show a statistically significant difference.

The author noticed that most participants simply selected the extreme option that described the operator positively. Participants were notified that the results were anonymous but there is still a possibility that participants got too involved with the operator during the interaction to be willing to give her a "bad" rating in the questionnaire for fear of letting her down or disappointing her. Possibly more care should have been taken to explain that this was not a personal measure of the operator but more so an evaluation the system. This was a complicated boundary to manage.

Cooperation was measured by three questions, this measure had the least number of questions to back it up and should therefore be interpreted more as an impression or indicator than a tried and tested statistic. Strong statistical difference was measured in favor of the expressive case. Cooperation in this sense means the willingness to cooperate with the robot-mediated partner as well as the perceived willingness of the partner to cooperate.

Behavior that is descriptive of an authoritative figure like a policeman or superior at work or a parent is usually a firm, assertive and static posture while that of a peer, be it a friend, a coworker or a sibling is usually more animated and playful. This might affect the perceived hierarchy between the robot-mediated partner and the participant and make the static robot look more authoritative while the expressive robot could be perceived more as a collaborator or peer.

Hypothesis number six was upheld but number four was not.

3.5.3 Engagement and Quality of Interaction

General Engagement

General engagement was measured by five questions, most of which showed a significant difference between the conditions as well as the averaged measure. Engagement is an important measure for formal and collaborative meetings but it is vital for any personal communication between families, spouses or friends. As family communication is a vital future application of telepresence systems, it pleased the author to see that the MeBot system helps engage people in the interactions on a deeper level than static systems.

Hypothesis number five was upheld.

Self-Similarity

One question asked for "How similar were the other person's suggestions to your own suggestions" ($p < 0.02$). This result might stem from similar reasons as were discussed in section 3.5.2. That is the robot-mediated partner was perceived as a peer when she was expressive. This result fortifies that assumption.

Competitiveness and Comfort

Competitiveness was measured by two questions, "How competitive was the other person?" ($p < 0.035$) and "How competitive were you?" ($p < 0.043$). Both show statistical significance in favor of the static condition, that is to say that participants that experienced the static condition reported that they themselves were more competitive as well as perceiving their partner to be more competitive.

Comfort was measured by two questions, "How comfortable did you feel during this activity?" ($p < 0.020$) and "How comfortable did the other person seem to be feeling during this activity?" ($p < 0.082$). A significant difference was found in favor of the expressive case. The results in the measurements of competitiveness and comfort both support our hypothesis that social expressiveness establishes flat hierarchy even further, as was discussed in the previous sections 3.5.2 and 3.5.3.

Enjoyment

Participants that experienced the expressive condition reported that they enjoyed using the system better ($p < 0.008$), this question showed strong statistical significance much to the author's content. It seems that people find an expressive robot more enjoyable than a passive one.

Hypothesis number seven was upheld.

Chapter 4

Conclusion

4.1 Conclusion

Akin, Minsky, Thiel, and Kurtzman stated in [Akin *et al.*, 1983] (1983) that telepresence is achieved when the following conditions are satisfied.

> "At the worksite, the manipulators have the dexterity to allow the operator to perform normal human functions. At the control station, the operator receives sufficient quantity and quality of sensory feedback to provide a feeling of actual presence at the worksite"

It is likely that they were talking about remote manipulators for industrial applications given the time at which this was published, but their definition can rather easily be extended to cover telepresence robotics for socially expressive communication. Effort was made to design a robot and a system that fulfilled these requirements as they apply to communication applications.

Systems Developed

A telepresence robot that allows for social expression was designed and built (2.2). A highly iterative design process was used and the robot's design went through about four major revisions and prototypes, taking into account lessons learned at every step and advancing the system towards the goal. Many supporting systems were developed along the way. Mechanical systems include a custom camera-embedded display (2.5.2) and a sympathetic controller (2.3.5). Electronic design includes sensory systems and acquisition techniques (2.2.5), motor control schemas which also underwent iterations of design (2.4), and custom control boards (2.2.5). Lastly the software design entailed developing customized audio/video transmission with face tracking (2.5.1), controllers that allow direct control of the robot's head movement with own head movement (2.3.4) and many other interfaces for control.

Evaluation Results

An evaluation of the system is presented. Participants performed a modified version of the *desert survival task* with a trained operator via the MeBot platform. Two conditions were compared, a mobile robot in a static pose and a mobile robot with expressive movement. A questionnaire was used to measure several dependent variables.

The expressive condition of the robot was ranked higher with statistically significant differences in all of the following measures: *psychological involvement*, *behavioral engagement*, *cooperation*, *general engagement* and *enjoyment*. An interesting result showed that participants that experienced the static condition reported more competitiveness both on their own part and the operator's with a statistically significant difference.

Contributions

In the field of telepresence, building socially expressive telerobots for a more effective, engaging and entertaining interaction is a novel idea. The platform shows promise of improved communication systems based on social expression, this needs further research. Technical contributions include a motor control scheme that is flexible, distributed and scalable for different applications while still providing structure for advanced control. Multiple novel interfaces were designed for intuitive control of the robot, including the use of head pose estimation to directly control the robot's head with your own head movements.

4.2 Future Work

In the previous chapters we showed that non-verbal expression can give rise to higher levels of engagement, psychological involvement, cooperation and enjoyment in people interacting with telerobots. Our experiment only begins to answer questions about expressive telerobots though and more research is needed to find out what constitutes adequate embodiment, which parts of expressivity play the most important role, the effects of representing a person using a non-human form of smaller scale etc.

We are very interested in executing another run of the same study that was performed for this thesis. That run would only test one condition which would be compared with data from the two conditions presented in this thesis. The condition would be a non-robot condition where participants would interact with their remote partner only through the OQO device, using audio and video. This would hopefully provide us with a base line for how much the embodiment of a robot affects how participants perceive the remote person, regardless of expressive movement.

The design of the MeBot is an actively growing project and we have several ideas for how to extend this project and further its development. Ideas that we want to focus on for future versions of the robot include the following:

- **Autonomous Navigation:** In effort to minimize the cognitive load on the operator we would like to implement an autonomous navigation system that, at least partially, relieves the operator from driving the robot around. The system would accept target positions from the operator and try to navigate the robot to that location the best it can. This system would have to perform statistical localization estimation as odometry data is noisy, and it would have to perform obstacle avoidance so not to drive the robot into items on the table.

- **Conversational Navigation:** From early stages of this project we had thought about designing a system that leverages face detection technology and the localization of the robot to autonomously manage multiple conversations for the operator. The system would recognize an interaction as a conversation and mark the robot's position and orientation in front of the person in question, it would do the same for all other interactions and provide the operator with an interface where they could simply select which person they want to speak with and the robot would autonomously navigate into position. The system would essentially maintain a human-occupancy map and continuously update it with information from sensors.

- **Peripheral Vision:** We would like to equip the robot with more sensors to provide the operator with better situational awareness. Much like humans have mostly motion detection activated for the periphery of their vision, we would like to make up for the robot's limited FOV by adding pyroelectric sensors that detect motion. The sensors would monitor the robots surroundings and alert the operator if there is significant movement suggesting that they should move the robots head towards it.

- **Stereo Microphones:** We think it could be crucial for operators that are participating in crowded meetings remotely to have stereo audio available to them. If anybody in the room would call to the operator to get their attention, it would be slightly awkward if the operator didn't know which direction to turn to face the person who called.

- **Postural Mirroring:** We are excited to investigate the effects of postural mirroring as it could be implemented as an autonomous behavior for the robot. Postural mirroring has been shown to be a good indicator of involvement and interest in personal communication ([LaFrance, 1985], [LaFrance & Broadbent, 1976]).

Appendix A

Study Material

This section contains some of the study material, namely the item list and the questionnaire. Two item list forms were presented to the participants, an *initial selections* form and a *final selections* form. The forms were identical except for their titles.

The questionnaire is composed of four parts, each one meant to measure different dependent variables. The following list explains the source of every part of the study:

- **PART I:** This part is composed of three subgroups of questions, measuring *co-presence*, *psychological involvement* and *behavioral engagement* respectively. Source [Biocca *et al.*, 2001].

- **PART II:** This part measures *trust*. Source Wheeless, L.R. and GROTZ, J. Individualized Trust Scale as reported in [Rubin *et al.*, 2004], page 184.

- **PART III:** The first subgroup in this part measures *general engagement*. Source measures of the six aspects of presence in [Lombard *et al.*, 2000]. The second subgroup measures a variety of things and was adapted from questions used for [Takayama *et al.*, 2009] and some new ones.

- **PART IV:** This part was intended to get demographic information from the participants as well as determine their familiarity with computers, robotics and videoconferencing.

DESERT SURVIVAL TASK
INITIAL CHOICE OF ITEMS

Please select the item from each pair that you would rather have if you were lost in the Desert.

1. (	) CANVAS	vs.	( ) TARP
2. (	) KNIFE	vs.	( ) PISTOL
3. (	) CHOCOLATE	vs.	( ) WATER
4. (	) MIRROR	vs.	( ) COMPASS
5. (	) FLASHLIGHT	vs.	( ) MATCHES
6. (	) AIR MAP	vs.	( ) BOOK ON EDIBLE ANIMALS

MEBOT QUESTIONNAIRE

Thank you very much for completing the interaction with the MeBot and for agreeing to complete this questionnaire.

Filling out this questionnaire is completely voluntary. You may choose to not answer any or all of the questions. Any answers you do provide will be kept strictly confidential and used only in evaluating our research.

There are four parts to this form, asking about the interactions that you just experienced and some general information about yourself.

INSTRUCTIONS:

The questions on these pages ask about the interactive experience you just had.

There are no right or wrong answers, please simply give your first impressions and answer the questions as accurately as possible, even questions that may seem unusual or to not apply.

Pease circle the responses that best represent your answers. All of your responses will be kept strictly confidential.

If you have any questions at any time while you are completing this questionnaire, please ask the experimenter for further explanation.

PART I

Please circle the number that represents your level of agreement with each statement as follows:

1 = Strongly Disagree
2 = Disagree
3 = Somewhat Disagree
4 = Neutral
5 = Somewhat Agree
6 = Agree
7 = Strongly Agree

PART I - A	Strongly Disagree						Strongly Agree
I often felt as if I was all alone	1	2	3	4	5	6	7
I think the other individual often felt alone	1	2	3	4	5	6	7
I hardly noticed another individual	1	2	3	4	5	6	7
The other individual didn't notice me	1	2	3	4	5	6	7
I was often aware of others in the environment	1	2	3	4	5	6	7
Others were often aware of me in the room	1	2	3	4	5	6	7

PART I - B	Strongly Disagree						Strongly Agree
I sometimes pretended to pay attention to the other individual	1	2	3	4	5	6	7
The other individual sometimes pretended to pay attention to me	1	2	3	4	5	6	7
The other individual paid close attention to me	1	2	3	4	5	6	7
I paid close attention to the other individual	1	2	3	4	5	6	7
My partner was easily distracted when other things were going on around us	1	2	3	4	5	6	7
I was easily distracted when other things were going on around me	1	2	3	4	5	6	7
The other individual tended to ignore me	1	2	3	4	5	6	7
I tended to ignore the other individual	1	2	3	4	5	6	7
When I was happy, the other was happy	1	2	3	4	5	6	7
When the other was happy, I was happy	1	2	3	4	5	6	7

The other individual was influenced by my moods	1	2	3	4	5	6	7
I was influenced by my partner's moods	1	2	3	4	5	6	7
The other's mood did NOT affect my mood/emotional-state	1	2	3	4	5	6	7
My mood did NOT affect the other's mood/emotional-state	1	2	3	4	5	6	7
My opinions were clear to the other	1	2	3	4	5	6	7
The opinions of the other were clear	1	2	3	4	5	6	7
My thoughts were clear to my partner	1	2	3	4	5	6	7
The other individual's thoughts were clear to me	1	2	3	4	5	6	7
The other understood what I meant	1	2	3	4	5	6	7
I understood what the other meant	1	2	3	4	5	6	7

PART I - C	Strongly Disagree						Strongly Agree
My actions were dependent on the other's actions	1	2	3	4	5	6	7
The other's actions were dependent on my actions	1	2	3	4	5	6	7
My behavior was in direct response to the other's behavior	1	2	3	4	5	6	7
The behavior of the other was I direct response to my behavior	1	2	3	4	5	6	7
What the other did affected what I did	1	2	3	4	5	6	7
What I did affected what the other did	1	2	3	4	5	6	7
My partner did not help me very much	1	2	3	4	5	6	7
I did not help the other very much	1	2	3	4	5	6	7
My partner worked with me to complete the task	1	2	3	4	5	6	7
I worked with the other individual to complete the task	1	2	3	4	5	6	7
The other could not act without me	1	2	3	4	5	6	7
I could not act with the other	1	2	3	4	5	6	7

PART II

Please select a rating based on what you feel best characterizes your partner

Trustworthy	1	2	3	4	5	6	7	Untrustworthy
Distrustful of this person	1	2	3	4	5	6	7	Trustful of this person
Confidential	1	2	3	4	5	6	7	Divulging
Exploitive	1	2	3	4	5	6	7	Benevolent
Safe	1	2	3	4	5	6	7	Dangerous
Deceptive	1	2	3	4	5	6	7	Candid
Not Deceitful	1	2	3	4	5	6	7	Deceitful
Tricky	1	2	3	4	5	6	7	Straightforward
Respectful	1	2	3	4	5	6	7	Disrespectful
Inconsiderate	1	2	3	4	5	6	7	Considerate
Honest	1	2	3	4	5	6	7	Dishonest
Unreliable	1	2	3	4	5	6	7	Reliable
Faithful	1	2	3	4	5	6	7	Unfaithful
Insincere	1	2	3	4	5	6	7	Sincere
Careful	1	2	3	4	5	6	7	Careless

PART III

Please circle the number that best represents your answer to each question

PART III - A

How engaging was the interaction?

Not at all engaged	1	2	3	4	5	6	7	Extremely Engaged

How relaxing or exciting was the experience?

Very Relaxing	1	2	3	4	5	6	7	Very Exciting

How completely were your senses engaged?

Not at all engaged	1	2	3	4	5	6	7	Extremely Engaged

The experience caused real feelings and emotions for me

Strongly Disagree	1	2	3	4	5	6	7	Strongly Agree

I was so involved that I lost track of time

Strongly Disagree	1	2	3	4	5	6	7	Strongly Agree

PART III - B

How well did you and the other person work together?

Not well at all	1	2	3	4	5	6	7	Extremely well

How much did you cooperate with the other person?

Barely cooperated	1	2	3	4	5	6	7	Cooperated a lot

How much did the other person cooperate with you?

Barely cooperated	1	2	3	4	5	6	7	Cooperated a lot

I really wanted to reach agreement with the other person

Strongly Disagree	1	2	3	4	5	6	7	Strongly Agree

How competitive was the other person?

Not competitive at all	1	2	3	4	5	6	7	Extremely Competitive

How competitive were you?

Not competitive at all	1	2	3	4	5	6	7	Extremely Competitive

How similar were the other person's suggestions to your own suggestions?

Not Similar at all	1	2	3	4	5	6	7	Extremely Similar

How comfortable did you feel during this activity?

Not comfortable at all	1	2	3	4	5	6	7	Extremely comfortable

How comfortable did the other person seem to be feeling during this activity?

Not comfortable at all	1	2	3	4	5	6	7	Extremely comfortable

PART III - C

How much enjoyable was your experience with the MeBot system?

Not enjoyable at all	1	2	3	4	5	6	7	Extremely enjoyable

I found that the robot's arms movement distracted me from the interaction

Strongly Disagree	1	2	3	4	5	6	7	Strongly Agree

I found that the robot's neck and head movement distracted me from the interaction

Strongly Disagree	1	2	3	4	5	6	7	Strongly Agree

I found that the robot's motor noise distracted me from the interaction

Strongly Disagree	1	2	3	4	5	6	7	Strongly Agree

What features/properties of the current implementation of the robot did you particularly like/dislike

____

____

____

What features/properties do you think would be most important in the future design of a system like the MeBot

____

____

____

PART IV

Please select the option that best describes you

How old are you (in years)? _________

Please indicate your gender

____ Male
____ Female

What is your race?

____ Asian
____ African American
____ Hispanic
____ Pacific Islander
____ Caucasian
____ Other – Please specify _________________
____ Multiracial

What is your level of education?

____ Some high school
____ High school degree
____ Some college
____ College degree
____ Some graduate school
____ Graduate school degree

How much do you know about robotics?

Nothing 1 2 3 4 5 6 7 A lot

How much do you know about computers?

Nothing 1 2 3 4 5 6 7 A lot

How many video-chats/meetings do you have in a month?

____ Zero
____ Less than once a month
____ 1-4 times a month
____ 5-10 times a month
____ 11-20 times a month
____ More than 20 times a month

Bibliography

[Akin *et al.*, 1983] Akin, DL, Minsky, ML, Thiel, ED, & Kurtzman, CR. 1983. Space Applications of Automation, Robotics and Machine Intelligence Systems(ARAMIS), phase 2. Volume 1: Telepresence technology base development[Final Report].

[Bailenson *et al.*, 2005] Bailenson, J.N., Swinth, K., Hoyt, C., Persky, S., Dimov, A., & Blascovich, J. 2005. The independent and interactive effects of embodied-agent appearance and behavior on self-report, cognitive, and behavioral markers of copresence in immersive virtual environments. *Presence: Teleoperators & Virtual Environments*, **14**(4), 379–393.

[Bente *et al.*, 2004] Bente, G., Rüggenberg, S., & Krämer, NC. 2004. Social presence and interpersonal trust in avatar-based, collaborative net-communications. *In: 7th Annual International Workshop on Presence.*

[Biocca *et al.*, 2001] Biocca, F., Harms, C., & Gregg, J. 2001. The networked minds measure of social presence: Pilot test of the factor structure and concurrent validity. *Pages 21–23 of: Presence 2001, 4th Annual International Workshop on Presence.*

[Brady, 2001] Brady, K., Tarn T.J. 2001. Handling Latency in Internet-Based Teleoperation. *Beyond webcams: an introduction to online robots.*

[Breazeal, 2003] Breazeal, C. 2003. Toward sociable robots. *Robotics and Autonomous Systems*, **42**(3-4), 167–175.

[Breazeal, 2004] Breazeal, C.L. 2004. *Designing sociable robots.* The MIT Press.

[Caldarelli & Campanella, 2003] Caldarelli, David D., & Campanella, Ruth S. 2003. *Ear.* World Book Online Americas Edition.

[Codourey *et al.*, 1997] Codourey, A., Rodriguez, M., & Pappas, I. 1997. A task-oriented teleoperation system for assembly in the microworld. *Pages 235–240 of: Proc. 8th International Conference on Advanced Robotics, 1997. ICARŠ97.*

[Fisch *et al.*, 2003] Fisch, A., Mavroidis, C., Melli-Huber, J., & Bar-Cohen, Y. 2003. Haptic devices for virtual reality, telepresence, and human-assistive robotics. *Biologically Inspired Intelligent Robots*, 73.

[Fong *et al.*, 2003] Fong, T., Nourbakhsh, I., & Dautenhahn, K. 2003. A survey of socially interactive robots. *Robotics and autonomous systems*, **42**(3-4), 143–166.

[Glas & Ishiguro, 2008] Glas, D.F., & Ishiguro, H. 2008. Simultaneous teleoperation of multiple social robots. *Pages 311–318 of: Proceedings of the 3rd ACM/IEEE international conference on Human robot interaction.* ACM New York, NY, USA.

[Goertz, 1952] Goertz, RC. 1952. Fundamentals of general-purpose remote manipulators. *Name: Nucleonics (US) Ceased publication.*

[Gu, 2008] Gu, Yingdan. 2008 (June). *Semi-Autonomous Mobile Phone Communication Avatar for Enhanced Interaction.* Undergraduate Thesis for the Department of Mechanical Engineering, MIT.

[Hancher, 2003] Hancher, Matt. 2003 (June). *A Motor Control Framework for Many-Axis Interactive Robots.* M.Eng. Thesis Department of Electrical Engineering and Computer Science.

[Hauber *et al.*, 2005] Hauber, Joerg, Regenbrecht, Holger, Hills, Aimee, Cockburn, Andrew, & Billinghurst, Mark. 2005. Social Presence in Two- and Threedimensional Videoconferencing. *In: PRESENCE 2005: The 8th Annual International Workshop on Presence.*

[Held & Durlach, 1991] Held, R., & Durlach, N. 1991. Telepresence, time delay and adaptation. *Pictorial communication in virtual and real environments*, 232–246.

[K. J. Åström, 1988] K. J. Åström, T.Hagglund. 1988. *Automatic tuning of PID controllers.* Instrument Society of America, (ISBN 1-55617-081-5).

[Karahalios & Dobson, 2005] Karahalios, K.G., & Dobson, K. 2005. Chit chat club: bridging virtual and physical space for social interaction. *Pages 1957–1960 of: Conference on Human Factors in Computing Systems.* ACM New York, NY, USA.

[Keyes, 2007] Keyes, B. 2007. Evolution of a telepresence robot interface. *Master's Thesis, Computer Science Department, University of Massachusetts, Lowell, May.*

[Keyes *et al.*, 2006] Keyes, B., Casey, R., Yanco, H.A., Maxwell, B.A., & Georgiev, Y. 2006. Camera placement and multi-camera fusion for remote robot operation. *In: Proceedings of the IEEE International Workshop on Safety, Security and Rescue Robotics.*

[Kidd & Breazeal, 2004] Kidd, CD, & Breazeal, C. 2004. Effect of a robot on user perceptions. *In: IEEE/RSJ International Conference on Intelligent Robots and Systems, 2004. (IROS 2004). Proceedings*, vol. 4.

[Koizumi *et al.*, 2006] Koizumi, S., Kanda, T., Shiomi, M., Ishiguro, H., & Hagita, N. 2006. Preliminary field trial for teleoperated communication robots. *Pages 145–150 of: Robot and Human Interactive Communication, 2006. ROMAN 2006. The 15th IEEE International Symposium on.*

[Lafferty & Elmers, 1974] Lafferty, J.C., Eady P.M., & Elmers, J. 1974. *The Desert Survival Problem.* Experimental Learning Methods.

[LaFrance, 1985] LaFrance, M. 1985. Postural mirroring and intergroup relations. *Personality and Social Psychology Bulletin*, **11**(2), 207.

[LaFrance & Broadbent, 1976] LaFrance, M., & Broadbent, M. 1976. Group rapport: Posture sharing as a nonverbal indicator. *Group & Organization Management*, **1**(3), 328.

[Lane & Akin, 2001] Lane, J. C., Carignan C. R., & Akin, D. L. 2001. Effects of Time Delay and Manipulator Speed on a Telerobotic Peg-in-Hole Task. 287–291.

[Li *et al.*, 1996] Li, L., Cox, B., Diftler, M., Shelton, S., & Rogers, B. 1996. Development of a telepresence controlled ambidextrous robot for space applications. *Pages 58–63 of: IEEE International Conference on Robotics and Automation*.

[Lombard *et al.*, 2000] Lombard, M., Ditton, T.B., Crane, D., Davis, B., Gil-Egui, G., Horvath, K., Rossman, J., & Park, S. 2000. Measuring presence: A literature-based approach to the development of a standardized paper-and-pencil instrument. *In: Third International Workshop on Presence, Delft, The Netherlands*, vol. 240. Citeseer.

[Marti & Schmandt, 2005] Marti, S., & Schmandt, C. 2005. Physical embodiments for mobile communication agents. *Pages 231–240 of: Proceedings of the 18th annual ACM symposium on User interface software and technology*. ACM New York, NY, USA.

[Michaud *et al.*, 2006] Michaud, F., Boissy, P., Corriveau, H., Grant, A., Lauria, M., Labonte, D., Cloutier, R., Roux, M.A., Royer, MP, & Iannuzzi, D. 2006. Telepresence robot for home care assistance. *Procceedings of AAAI*.

[Minsky, 1980] Minsky, Marvin. 1980. Telepresence. *OMNI magazine*.

[Negroponte, 1983] Negroponte, Nicholas. 1983. Talking Heads. *Discursions (Video Disc), Side One track 14*.

[Paulos & Canny, 1998] Paulos, E., & Canny, J. 1998. Designing personal tele-embodiment. *In: 1998 IEEE International Conference on Robotics and Automation, 1998. Proceedings*, vol. 4.

[Paulos & Canny, 2001] Paulos, E., & Canny, J. 2001. Social tele-embodiment: Understanding presence. *Autonomous Robots*, **11**(1), 87–95.

[Rubin *et al.*, 2004] Rubin, R.B., Palmgreen, P., Sypher, H.E., & Beatty, M. 2004. *Communication research measures: A sourcebook*. Lawrence Erlbaum.

[Sakamoto *et al.*, 2007] Sakamoto, D., Kanda, T., Ono, T., Ishiguro, H., & Hagita, N. 2007. Android as a telecommunication medium with a human-like presence. *Pages 193–200 of: Proceedings of the ACM/IEEE international conference on Human-robot interaction*. ACM New York, NY, USA.

[Stiehl *et al.*, 2006] Stiehl, WD, Lieberman, J., Breazeal, C., Basel, L., Cooper, R., Knight, H., Lalla, L., Maymin, A., & Purchase, S. 2006. The huggable: a therapeutic robotic companion for relational, affective touch. *In: 3rd IEEE Consumer Communications and Networking Conference, 2006. CCNC 2006*, vol. 2.

[Takayama *et al.*, 2009] Takayama, L., Groom, V., & Nass, C. 2009. I'm sorry, Dave: i'm afraid i won't do that: social aspects of human-agent conflict. *Pages 2099–2108 of: Proceedings of the 27th international conference on Human factors in computing systems*. ACM New York, NY, USA.

[Tojo *et al.*, 2000] Tojo, T., Matsusaka, Y., Ishii, T., & Kobayashi, T. 2000. A conversational robot utilizing facial and body expressions. *In: 2000 IEEE International Conference on Systems, Man, and Cybernetics*, vol. 2.

[Tsai *et al.*, 2007] Tsai, T.C., Hsu, Y.L., Ma, A.I., King, T., & Wu, C.H. 2007. Developing a telepresence robot for interpersonal communication with the elderly in a home environment. *Telemedicine and e-Health*, **13**(4), 407–424.

[Yoerger & Slotine, 1987] Yoerger, D., & Slotine, J.J. 1987. Supervisory control architecture for underwater teleoperation. *In: 1987 IEEE International Conference on Robotics and Automation. Proceedings*, vol. 4.