INTERACTIVE PHYSICAL AGENTS FOR STORY GATHERING

by
Alexander James Reben

Bachelor of Science in Applied Math
State University of New York at Stony Brook, 2003

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

Master of Science in Media Arts and Sciences

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 2010

Signature of Author ____

Program in Media Arts and Sciences
May 14, 2010

Certified by ____

Joseph A. Paradiso
Associate Professor of Media Arts and Sciences
MIT Media Laboratory
Thesis Supervisor

Accepted by ____

Prof. Pattie Maes
Associate Academic Head
Program in Media Arts and Sciences

INTERACTIVE PHYSICAL AGENTS FOR STORY GATHERING

by
Alexander James Reben

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

Abstract

Robots are typically thought of as autonomous devices which require little to no human interaction to complete their goals. In this study we investigated what would happen if the success of a robot was contingent upon its interaction with a human. How do we leverage humans to make robots more intelligent, efficient and successful? Is it worth it to involve human heavily in the goals of a system? This thesis we presents a method for the creation of a physical agent to facilitate interaction and documentary gathering within a ubiquitous media framework as a method for studying human-dependent systems. We built a robot and sent it out to autonomously capture stories about its environment. The robot had a specific story capture goal and leveraged humans to attain that goal. The robot gathered a 1st person view of stories unfolding in real life. We evaluated this agent by way of determining "complete" vs. "incomplete" interactions. "Complete" interactions were those that generated viable and interesting videos, which could be edited together into a larger narrative. It was found that 30% of the interactions captured were "complete" interactions. Results suggested that changes in the system would only produce incrementally more "complete" interactions, as external factors like natural bias or busyness of the user come into play. The types of users who encountered the robot were fairly polar; either they wanted to interact or did not - very few partial interactions went on for more than 1 minute. Users who partially interacted with the robot were found to treat it rougher than those who completed the full interaction. It was also determined that this type of limited-interaction system is best suited for short-term interactions. At the end of the study, a movie was produced from the videos captured, proving that they were viable for story-making.

Thesis Supervisor: Joseph A. Paradiso

Title: Associate Professor of Media Arts and Sciences, MIT Media Lab

INTERACTIVE PHYSICAL AGENTS FOR STORY GATHERING

by
Alexander James Reben

The following people served as readers for this thesis:

Thesis Reader

Dr. Cynthia Breazeal
Associate Professor of Media Arts and Sciences
Co-Director, Center for Future Storytelling
Director of the Personal Robotics Group
LG Group Career Development Professor

Thesis Reader

Randy Sargent
Senior Systems Scientist at Carnegie Mellon University

Acknowledgments

This thesis has proven to be both challenging and rewarding, it has changed my approach to research and helped me grow as a researcher. I would not have been able to do it without the help of a multitude of people. Foremost, a thank you goes out to my advisor Joseph Paradiso who supported my ideas and provided guidance during my time at the Media Lab. I'd like to thank my readers Cynthia Breazeal and Randy Sargent. Cynthia has proven an invaluable resource for ideas and current research in the HRI field. Randy Sargent, whom I worked with at NASA, for encouraging me to come to the Media Lab in the first place and for being an inspiration to push myself harder.

A big thank you to the responsive environments group:

Nan-Wei Gong: For being an outstanding officemate, a positive influence when things get hard, wrangling SPINNER and being easily available when I need to annoy someone.

Mat Laibowitz: For always lending a helping hand, providing a steady stream of advice, keeping a steady supply of po-stash around and expanding my Italian film literacy.

Matt Aldrich: For getting in there to peep anything I needed, helping me figure out the main move when I needed to deal with filth, going to the trucks with me to get a sando, expanding my vocabulary and hosting many hang-out sessions. (yoink)

Bo Morgan: For keeping the SPINNERD torch burning.

Mike Lapinski: For being such a Ron and giving me the opportunity to work with the Red Sox.

Mark Feldmeier: For being a living transistor man and constantly making sure all is well with me.

Gershon Dublon: For making sure there is always some strange instrument playing behind me.

Laurel Smith Pardue: For holding cool and crazy concerts.

Nan Zhao: For being a cool German and hosting get-togethers.

I'd also like to thank my UROP Jeremy Kuempel who produced robot designs for the thesis and helped complete many tasks, my parents for helping me get to where I am today, Aithne Sheng-Ying Pao for her support and encouragement, group administrators Lisa Lieberson and Amna Carreiro, and the many other great people I have met during my time at the Media Lab.

Table of Contents

Figures and Tables

Figures:

Tables:

Chapter 1

Introduction

1.1 Motivation

Robots are typically thought of as autonomous devices which require little to no human interaction to complete their goals. In this study we investigated what would happen if the success of a robot was contingent upon its interaction with a human. How do we leverage humans to make robots more intelligent, efficient and successful? Is it worth it to involve human heavily in the goals of a system? One system type in which we saw room for such improvements was media capture. Currently, distributed media capture systems, such as security cameras, are typically passive, and don't actively engage their subjects or try to actively extract a narrative or documentary from them. Aside from simple panning/tilting, most ubiquitous media capture facilities also tend to be restricted to fixed viewpoints. For example, systems such as the Media Lab's ubiquitous media portals can only capture stationary 3rd person views of the environment [1]. While this may work for many situations, a mobile system node would allow for first person capture of narrative content and dynamic sensor placement. If a human is actively engaged in the capture loop, a more intelligent capture system can be implemented that may be more effective at generating meaningful documentaries, provided the user can be somehow persuaded to help the system achieve its goal.

1.2 Purpose and Goals

The purpose of this thesis was to create a physical agent to facilitate interaction and documentary gathering within a ubiquitous media framework as a method for studying human-dependent systems. We built a robot that had a goal of actively capturing a story about its environment and the people within it. That is, that the robot had a specific story capture goal and leveraged its mobility and interactivity together with the capabilities of the ubiquitous sensor network to achieve that goal. This device also allowed for an active first person view that many distributed media capture systems don't enable. It can also reach areas that 3rd person systems do not cover by its inherent mobility. In these blind areas the robot either recorded data there or prompted others to move it to an area where sensors are active. The novel approach to this system is that it brings activities of interest to the area where they can be most effectively captured by leveraging human empathy and interaction. Through this empathy and interaction this engagement encouraged the person interacting with the robot to share their story, in a meaningful way. We evaluated the effectiveness of different forms of interaction which were developed on the robot platform. This interaction design and testing in real world scenarios was the focus of this thesis. The robot acts as a facilitator to coax those who may not initially be inclined to interact with the system to share their stories. The documentary created started on the small scale, at the immediate level of the person interacting with the robot, and expanded through that interaction to encompass stories from others who subsequently encounter the robot.

A narrative "thread" followed the robot through its interactions and path of story goal achievement. Through this, we can see how the robot's users are connected within the story line, through the robot's interactions. This provides a way to incorporate social interaction within the framework of ubiquitous media systems. This system is the first we know of that has the specific goal of actively capturing a documentary within a ubiquitous media environment by leveraging human intelligence, interaction and emotion. This novel approach created a rich story capture system within and outside of ubiquitous media frameworks.

The goals of this thesis were to study the best interaction scenarios and methods with which to leverage human interactions into the creation of stories. This was to be accomplished in several ways. The system was evaluated through user trials. These trials were run with multiple participants interacting with the robots as they roam an area. The users would be random as the robot found them, and would not have knowledge of the robot beforehand. The criterion to be met for the system to be a success was the quality of narrative extraction from users and success of interactions. The interaction between the users and the robot was logged and conclusions were to be drawn from this interaction as to which type of interaction is most successful for a particular documentary capture scenario.

More specifically, the types of interactions and documentary capture types that were to be evaluated are as follows:

Interactions:

- Emotion (sympathy, humor) – The robot will act in a helpless or funny manor asking people to intervene.
- Gaming –The robot is part of a game in which people can participate.
- Simple instructions – The robot provides instructions to the user to carry out.

Documentary Capture Types:

- Interview (personal) – The robot gathers personal documentaries from those it interacts with.
- Location (public) – The robot gathers documentaries about a particular location through people.
- Social (inter-personal) – The robot gathers documentaries about social connections.

Quality of Interaction:

- Interaction length – How long the user interacted with the robot.
- Story quality – How well did the user share a story with the robot.

Chapter 2

Background

2.1 Related Work

2.1.1 Human-Robot Interaction

Almost every robot developed and deployed must interact with humans in some way. Robots can interact with humans on several levels, which can range from minimal maintenance interaction to interaction mimicking that of a person to person conversation. The robot interaction must be custom fit to the particular robot task and goal set. One might not want their industrial robot to talk back to them, but having a voice may be essential to robots in non-formal and social contexts. If the interaction is ill-fit to the particular task, the robot may not be able to carry out its functions and goals effectively, if at all. This thesis dealt with robots based in the higher interaction category. This category involves robots which interact with humans on a social level, so called "sociable robots".

Sociable robots can be defined as robots which leverage social interactions and cues in order to attain the internal goals of the robot [2]. Sociable robots interact with humans on their own terms, having to respect social norms and behaviors, those who interact with the robot need to identify and empathize with the robot. Sociable robots usually are anthropomorphic, having features the user can empathize and identify with

[3]. An example of a sociable robot is the Nexi robot at the MIT Media Lab. This robot has human features and a voice to communicate with the user (Figure 2.1).

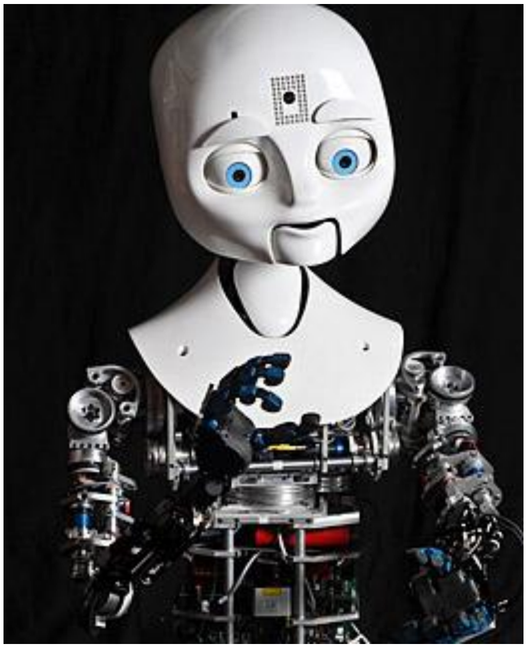


Figure 2.1: An example of a sociable robot. Nexi MDS robot developed at the MIT Media Lab.

It is sometimes dangerous not to consider the human aspect when developing an agent for a system. Take for example "The interactive museum tour-guide robot" experiment [4]. The team developing the system did not take into account how humans would react to the robot. They released their huge garbage-can like robot into a museum, with the intent to lead people around. However, they found that simply having the robot issue instructions to the users did not make them comply. For example, when the robot needed to move, it would check for a clear path and travel through that path. Problematically, the robot would be surrounded by humans most of the time, having no clear path. The researchers eventually added a voice asking people to move out of the way. However, once one person moved, another stepped in their place to take a look at the interesting robot. Finally, when they added a happy and sad face to the robot (the robot being sad when it could not move) people would move out of the robot's way to avoid upsetting it [4]. This clearly demonstrated that the interaction of a robot with its users is as important as any internal programming or engineering considerations for goal achievement.

Robots which require humans to complete their goals are what we call "human-dependent". Systems which are human-dependent require a mechanism to motivate humans to interact with the system. We believed that if an interaction with a robot is interesting and mimics social interactions, we would be able to more effectively carry out the goals of our system. In fact, the success of our robot is contingent upon humans interacting with it.

2.1.2 Leveraging Humans

An example of a system which leverages human mobility to move sensor nodes to specific locations is the "Parasitic Mobility" system. The "Parasitic Mobility" system relied on humans and other vehicles to spread sensor "parasites" to various locations (see Figure 2.2) [5]. The system latched onto the subject or created an instruction for them to carry out in order to leverage their mobility. The system also explored a symbiotic approach as a contrast to the parasitic mechanism. This "symbiotic mobility" approach attempted to offer the user utility for carrying out an action for the system. For example, the device would change color or change in some other interesting manor in exchange for the human completing an action (see Figure 2.3). This feedback was accomplished in several ways, such as if the user took the device to a wrong location, the device would buzz and vibrate annoyingly encouraging users to put the node down. This system differed from our system in that it does not have a story capture goal. It had a goal of moving the sensor node to a location of interest, whereas our system moves both the person and the robot to the location where the best capture can occur. Our system also had the ability to move on its own; if there are no users to interact with in the current space, it can seek out a new area.

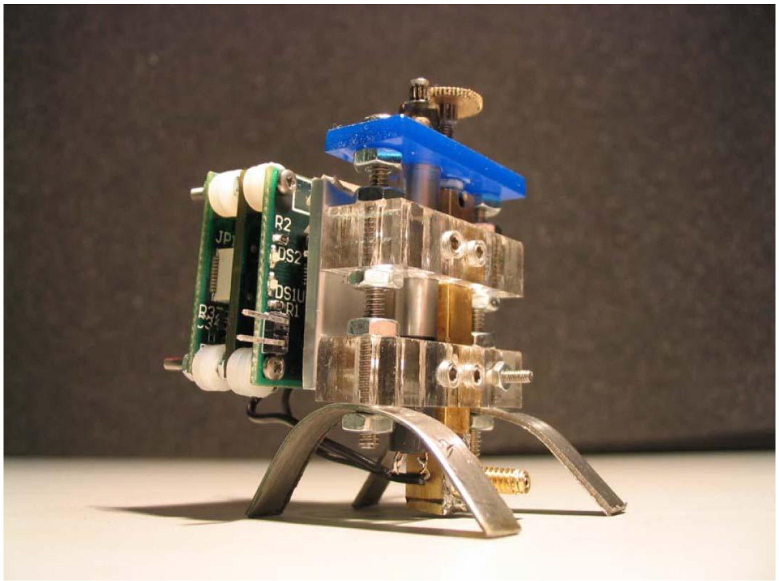


Figure 2.2: A sensor node which utilizes parasitic mobility.

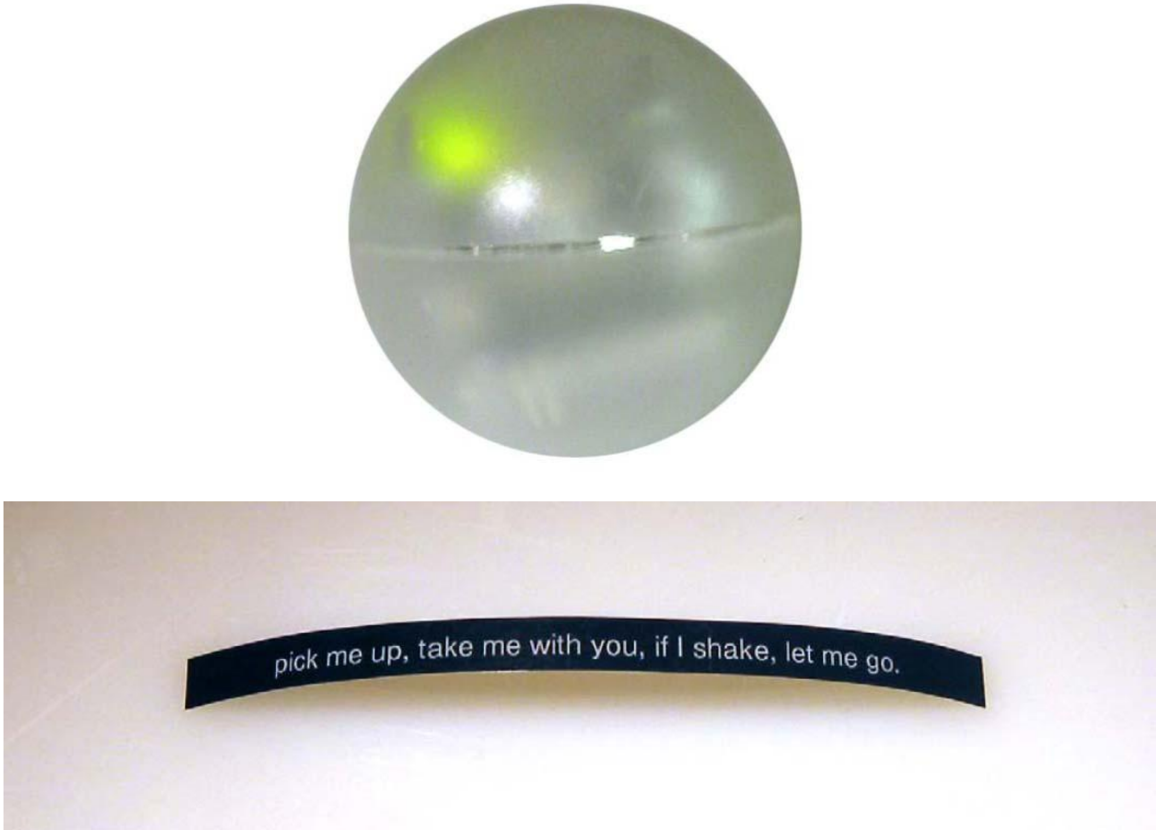


Figure 2.3: A sensor node which uses symbiotic mobility, along with its instruction tag.

An example of a mobile platform that relies upon human sympathy for goal attainment is the “tweenbot”. The tweenbot study was a major inspiration for this research and the development of the “Boxie” robot used in this thesis. The tweenbot is a robot shaped cardboard device that rolls in a straight line. Attached to it, is a flag with the device’s goal written on it. The device is set in a public place with the hopes of attaining the goal written on it. Since it only travels in a straight line, it completely relies on human intervention to achieve its goal. It was found that simply having the robot look lost and helpless encouraged those around it to help it towards its goal.

> *“The results were unexpected. Over the course of the following months, throughout numerous missions, the Tween bots were successful in rolling from their start point to their far-away destination assisted only by strangers.” [6]*

The tweenbot differs from Boxie in that the tweenbot’s only goal was to travel towards a location. It did not need mechanisms for extended interaction, as the extent of the interaction was for the user to point the robot in the correct direction. The tweenbot also did not produce anything from its interactions; it was simply a test to see if a device could navigate leveraging the kindness of humans. The Boxie robot recorded its interactions and extracted narrative from its users. A similar study was carried out by Wiess et al. In "Robots Asking for Directions – The Willingness of Passers-by to Support" a robot was put in a public area and was set to ask for directions. The robot interacted with a combination of speech output and a touch screen. The robot would ask for directions to a particular spot and users would point and input commands via the screen [7]. This system attempted to use a higher level interaction scheme than that of the tweenbot. There was little attention paid to the physical appearance of the robot, as it was assembled from stock components and garbage-can-like (much like the museum robot discussed earlier). This robot differs from the Boxie system in that it did not attempt to leverage human sympathy to attain its goals. It also did not pull any content from the users - like the tweenbot its main concern was to get to a location.

One robot which did produce a tangible output from its users was the "STB: human-dependent sociable trash box". The robot, which was shaped like a trash can, would rove around trying to elicit children to deposit trash in its receptacle. The robot could not complete its goal on its own without the interaction of the children. The robot utilized vocalizations and behaviors to try to elicit trash from the children. It was found that the robot could achieve its goal if the robot was moved toward an area with trash by the children [8].

2.1.3 Story Gathering

Systems which "story gather" attempt to pull stories out of the environment. They will either passive or actively attempt to gather stories. A passive story gatherer is like a security camera while an active story gatherer is like a reporter. Both types of story gathering techniques have their advantages and disadvantages.

In "Dolltalk: A computational toy to enhance children's creativity" a tool for encouraging children to tell and play-act stories was developed. The system used toys embedded with sensors to interact with children. The children formed personal connections with the toys, which encouraged them to tell stories. It was found that:

> *"Although children enjoyed the [story telling] activity, they appeared to need an audience, as a goal for the story, but also as guidance through the interaction."[9]*

This confirms the importance of interaction in the story gathering process. It also shows that a defined story goal is important in motivating story telling. Unlike the Boxie robot, the dolls used in this study did not move or try to engage those which were not immediately interacting with the device. This system was a hybrid of active and passive story gathering methods. Another example of a device to encourage storytelling was the Afghan Explorer developed at the Media Lab. This active "roving reporter" robot was intended to roam battlefields to collect news stories in lieu of a human reporter. Being that the robot was not human, the loss of the robot due to warfare would not have the same impact as a human fatality [10]. This platform aimed to demonstrate the utility of having an agent that could be independent of humans and gather stories on its own. This platform did not attempt to leverage human mobility. Instead it was sent into war zones with the hope that those who saw it would interact with it.

The network of sensor portals currently deployed at the Media Lab is an example of a ubiquitous media capture system. Laibowitz's SPINNER application that's currently running on the system passively captures events of interest when they occur. The SPINNER application automatically creates video narratives based on sensor inputs. It does this by leveraging the video-capturing Ubiquitous Media Portals distributed throughout the lab (see Figure 2.4). Within the system, badges can be used to mark participants within the range of the portal and record sensor data from their location. The SPINNER application then uses this labeling to determine which media clips best fit a specified narrative timeline, and assembles the clips accordingly to produce a video that best fits the input "story" [1]. While this application provides the basis of this thesis and supports it, it differs in several key aspects. Its current passiveness is not conducive to interaction. Because deriving narratives from sensor data is very challenging, SPINNER could only produce simple storylines such as action vs. inaction and social vs. non-social. Furthermore, it does not take an active approach to documentary gathering, which is only optimal for narratives in a 3rd person perspective. The Boxie robot augments and expands the current ubiquitous media portal system, providing a new level of interaction, a first person view and goal oriented story capture.

The SenseCam developed by Microsoft research is an example of a 1st person media and sensor capture system [11]. The SenseCam device is a "Retrospective Memory Aid". It consists of a still camera and sensor that is worn around the user's neck. The camera takes pictures and logs sensor data throughout the day and saves them for later review. Much like the SPINNER system, it relies on capture of events passively without interaction, and does not have a particular story capture goal. It also is a personal and symbiotic device, only capturing a story form one viewpoint.

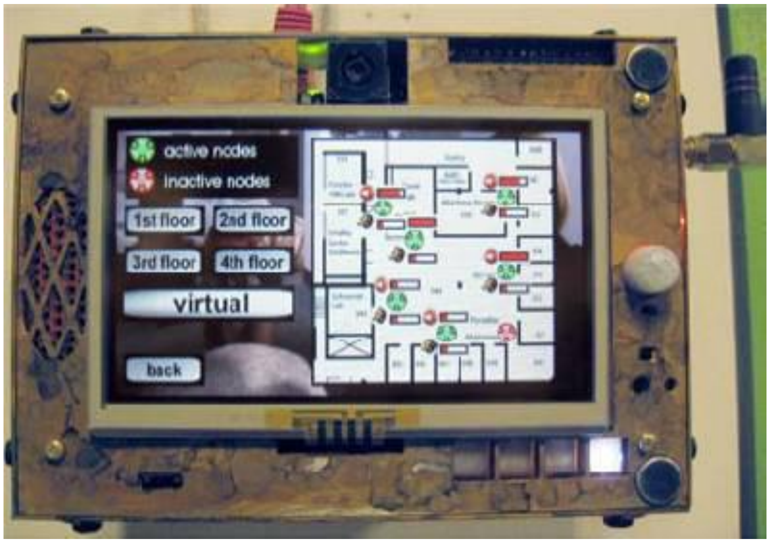


Figure 2.4: Ubiquitous media capture portal.

Chapter 3

Interaction Design

3.1 Robot Design Considerations

The first step in the design of a sociable robot which leverages humans is to consider the interaction with the user. There are several factors which determine the success of such a system. The factors for our particular system were overall “cuteness”, interaction simplicity, appearance and behavior. The overall interaction paradigm for the robot was that of an “active” story gathering robot. Bergström et al. determined that an active robot in a crowded public area (that of a shopping mall) was most successful in eliciting interactions with passers-by if it took an active role in the human acquisition process. It was also found that humans will adjust their positions in order to be “noticed” by the face of the robot.

“One interesting observation, more related to human behavior, is that the test subjects when standing seemed to adjust their positions in order to stand more face to face with the robot if they felt like they were being noticed.” [12]

The idea of a human being “noticed” was a central consideration in the interaction design of the robot. To be noticed, the robot needed a face and a gaze. It also needed to recognize the user and acknowledge their presence.

3.1.1 "Cuteness"

The main mechanism the robot uses to leverage humans is cuteness. Our emotional reactions and empathy for cute things is hardwired into our brains through evolution. The subject of cuteness has been long studied by evolutionary psychologists. The reaction we have to cute things originates with the need to care for babies [13]. The cuteness of babies motivates parents to care for and protect them. We used the cuteness of the Boxie robot to draw users in and keep them engaged. The cuteness paradigm we chose to use is that of the lost and helpless child. It was felt that if the robot was perceived as a helpless and lost "child-bot" that user's instincts would engage and they would interact with the robot until they were able to help it.

A major consideration was how to engage the user and keep them engaged. We needed to be careful not to make it seem as though the robot was using cuteness to exploit the user to achieve its own goals (even though that is the goal of the system).

"…the rapidity and promiscuity of the cute response makes the impulse suspect, readily overridden by the angry sense that one is being exploited or deceived." [14]

We avoided this pitfall by carefully molding the interaction with the robot to gradually ask the user to work more for the system. The robot also interacted in a two way fashion, giving the user something in-kind to their input. For example, when the user told the robot something about them, the robot would tell it something about itself. With this approach, we avoided the perception by the user that they were being used by the system, making the interaction enjoyable and effective.

There are several factors which make something cute. Physical characteristics include eye size, eye spacing and position, head size, body proportions. Behavior characteristics include naivety, awkwardness, childish sounds, non-efficient locomotion and helplessness [14-15]. We evaluated these parameters by creating physical prototypes and by testing various interaction scenarios.

3.1.2 Simplicity

Simple systems come with a myriad of benefits: low cost, low weight, less failure points, easy to maintain, clean aesthetics etc… Every development step of Boxie had simplicity in mind. The robot needed to be low cost, since the robot may be stolen. Its weight had to be kept low in order for users to be able to pick it up comfortably. Failure points had to be kept to a minimum and any failure had to be easily repairable. The appearance had to be robust and likable. All of these factors led to a better interaction experience for the user.

The interaction between the user and the robot was simple and intuitive. The input for the user was in the form of two simple pushbuttons on the side of the robot's head. One was green and the other was red (see Figure 3.1). The robot explained the function of the buttons to the user at the appropriate time. Since there was only one mode of input, the confusion for the user was kept at a minimum. Other inputs were transparent to the user, they were inferred from reading sensors on the robot. These included: Sensing people in the vicinity of the robot, sensing when the robot is picked up and put down, sensing how far the user is from the robot, sensing the tilt of the robot and sensing the location of the robot.

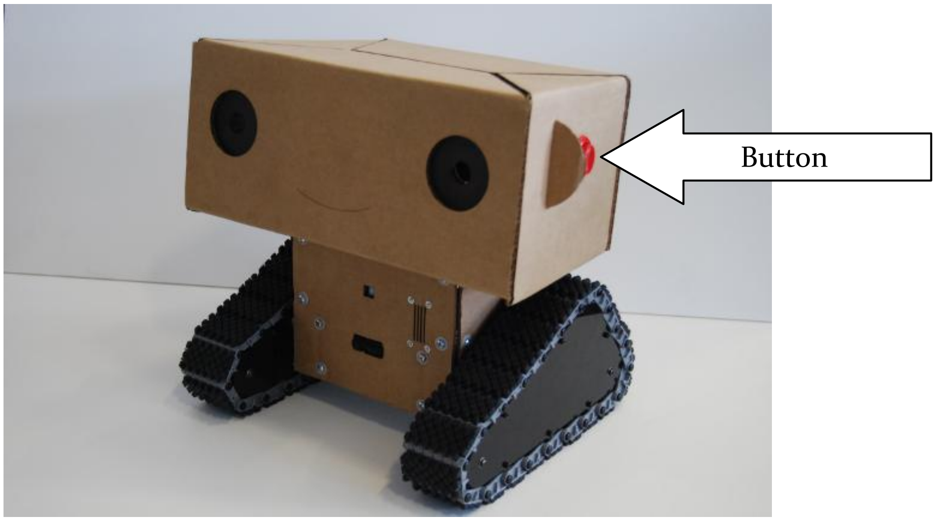


Figure 3.1: Button input on the side of the robot's head.

The output of the robot consisted of two mechanisms. One was the movement of the robot, the other was sound. The movement of the robot could be controlled in order to convey information about the robot. The robot could become stuck to look helpless, or look like it was searching for something to elicit help. The main output was the audio system. This system allowed the robot to speak and play sounds. Keeping the output systems simple allowed for a more understandable robot and kept distractions to a minimum.

3.1.3 Appearance

Appearance is an important factor in addressing interactivity and usability of a robot. Since the robot would be relying on its cuteness in order to leverage humans, special care was taken in the design of its appearance. The overall aesthetic of the robot was one of a box-shaped tracked robot. We chose this shape because of its simple lines and familiar appearance. It is also a shape often associated with robots appearing in science fiction. The first factor to be taken into consideration was the proportions of the major components. The major components were the head, torso and tracks. A minimal size for the components was determined by the largest internal components. The largest internal components were the video camera and battery. The video camera, which was mounted behind an eye, informed the height of the head. The battery mounted in the bottom (for low center of gravity stability purposes) informed the width of the torso. Starting from the minimum requirements, multiple designs were investigated.

SolidWorks was used as a modeling tool which allowed us to laser cut out of cardboard physical prototypes. This rapid design and prototype processes allowed us to evaluate multiple physical designs. The physical prototypes gave a better feel for the cuteness of the robot that was not adequately represented on screen.

The physical characteristics of most importance included eye size, eye spacing and position, head size, body proportions. The eyes of the robot were chosen to be large circular eyes. Geldart et al. determined that figures with larger eye sizes were found to be more attractive by both adults and children [16]. This eye geometry yielded the "cutest" looking results in the prototypes. The eyes were set far apart and low on the head, making the robot appear naive and young. The head was made large in relation to the torso, a 3:2 ratio was found to be most aesthetically pleasing. A short squat body was the most childlike configuration. We considered how the user would hold the robot. The only way to test how a user might approach holding the robot was to develop physical prototypes and ask users to grasp the robot in the most natural way.

Multiple prototypes were designed, built and evaluated to find the best combination of features and the best ratios. Figure 3.2 shows one of the first attempts to design the robot's appearance. The treads and base were predetermined designs, so the torso and head were the portions which were modified. In Figure 3.2 we can see a tall square torso with a square head. In this initial design we fleshed out the general aesthetics of the eyes, however the torso was not pleasing. The tall torso made the robot seem unbalanced and unwieldy.

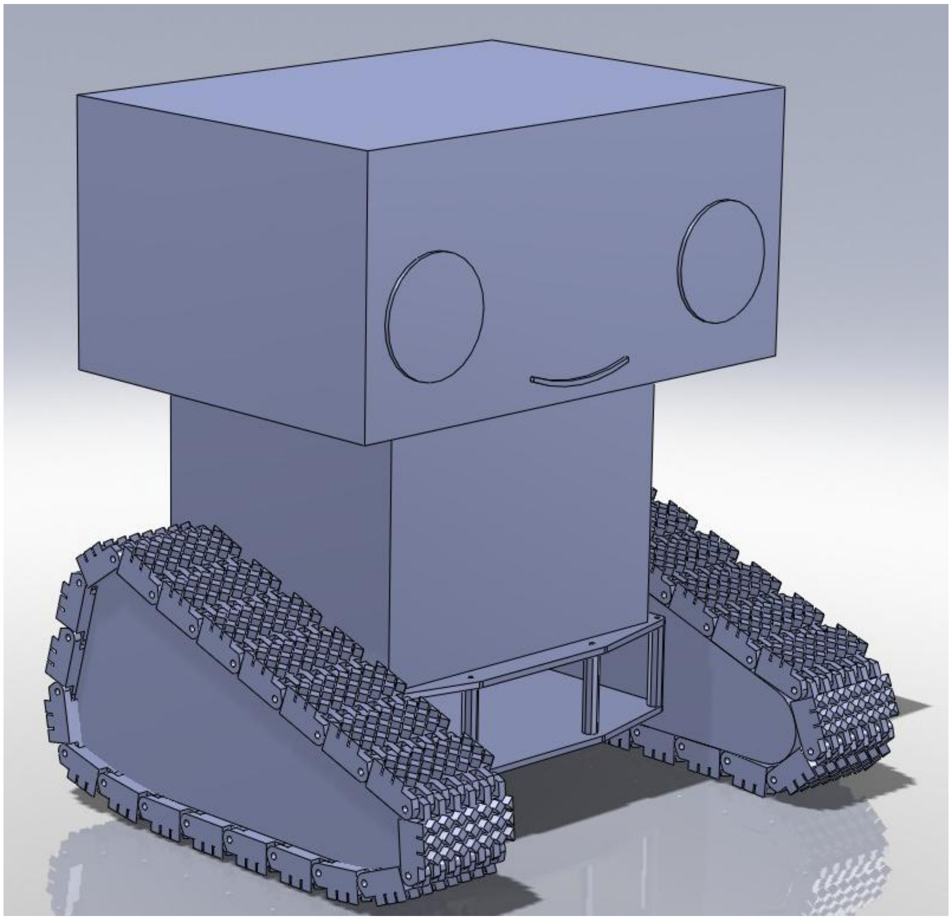


Figure 3.2: Robot with a square torso, square head. Initial design iteration in SolidWorks.

The second iteration, shown in Figure 3.3 shows a modification of the head. Instead of square, the head was tapered toward the back. The backward sweep made the head appear more human like, this slight modification increased anthropomorphic characteristics of the head.

Figure 3.4 introduces a pyramidal torso and Figure 3.5 shows other features such as uneven eyes and antennae. However, this design in prototyping was found not aesthetically pleasing and had issues with stability.

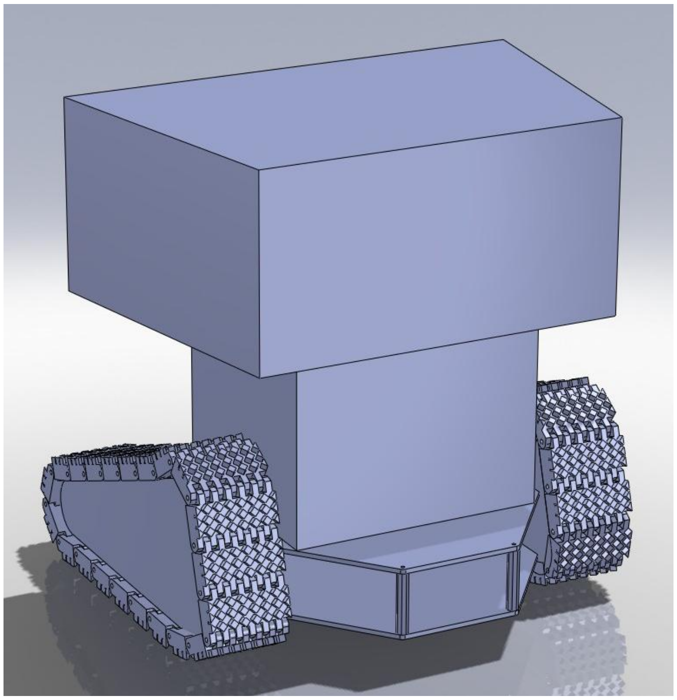


Figure 3.3: Robot with a tall square torso, trapezoidal head. The trapezoidal head made the robot take on a more anthropomorphic appearance. This slight design modification is an example of changes made after producing a physical prototype.

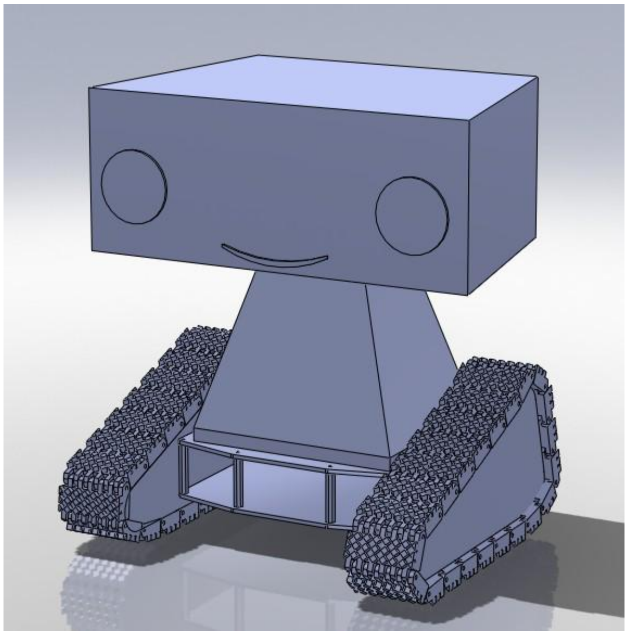


Figure 3.4: Robot with a pyramidal torso, trapezoidal head.

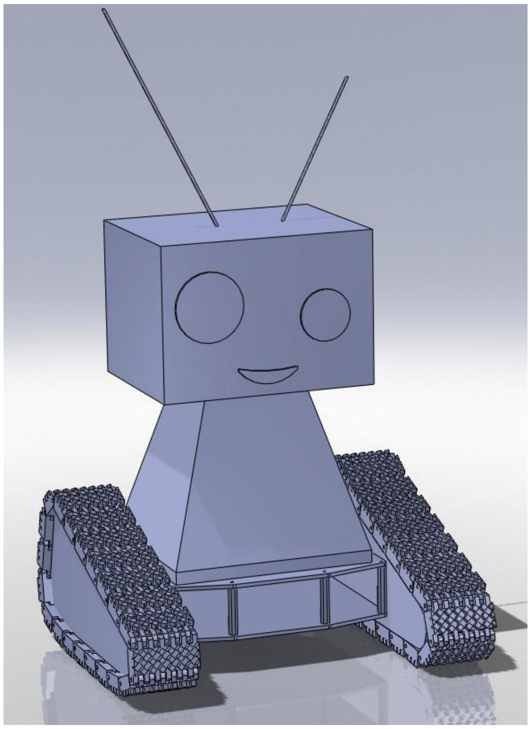


Figure 3.5: Robot with a pyramidal torso, square head, non-symmetrical eyes and antennae.

Figure 3.6 shows a squat robot with a square head which eventually evolved into the final prototype shown in Figure 3.7.

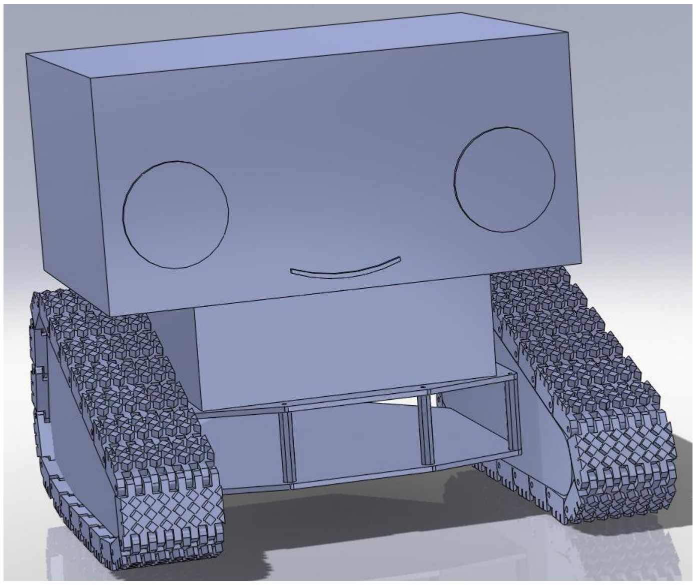


Figure 3.6: Robot with a squat square torso, square head.

Figure 3.7 combines the short squat torso, large symmetrical eyes and the trapezoidal head aspects of the other prototypes into one design. Taking the best features of the other prototypes, we produced the final design.

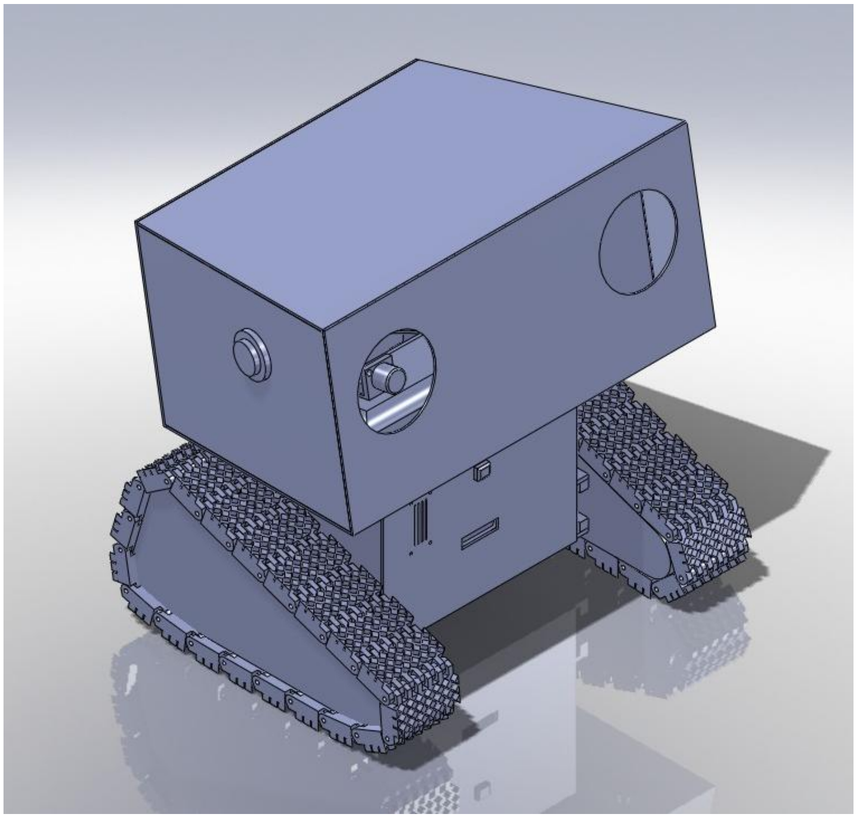


Figure 3.7: Robot with a square squat torso, trapezoidal head.

Figure 3.8 shows a typical cardboard prototype of a design. The cardboard panels were laser cut and assembled with glue to get a real word feel of the prototype. This method was extremely valuable as the 3D rendering in SolidWorks did not fully represent the aesthetic of the robot and 3D models on a 2D screen did not offer optimal perception of the design. Having a tangible model to hold and move allowed us to better implement future design iterations and quick find and fix problems.

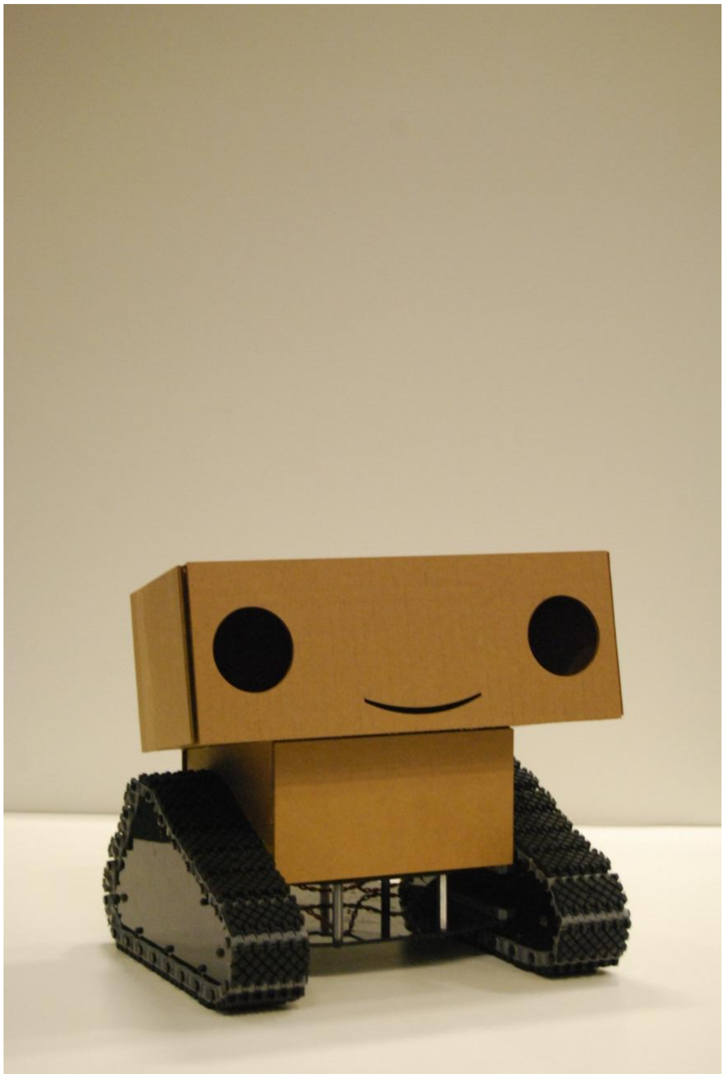


Figure 3.8: Prototype of Figure 3.6, assembled from laser cut cardboard panels and glue, set atop the tread base made from pre fabricated treads and laser cut acrylic base plates.

Originally, the intent was to make the robot out of translucent white acrylic. However, after the acrylic shell was fabricated, the general consensus was that the cardboard prototypes we had been producing looked “softer” and “more organic”. The acrylic shell appeared industrial and frightening (see Figure 3.9). In testing, it was also found that the cardboard construction was more robust. This was due to the brittle nature of acrylic verses the more forgiving cardboard (see section 4.1.3 for more discussion regarding the mechanical design of the shell).

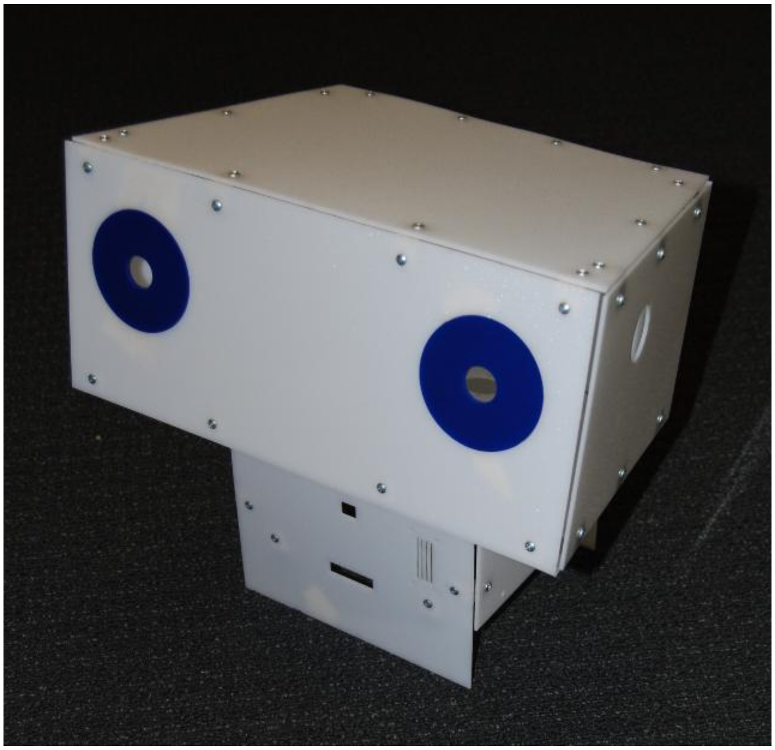


Figure 3.9: Acrylic robot fabricated.

Figure 3.10 shows the fully assembled robot, which is modeled in Figure 3.7. This design was chosen as the best compromise of structural stability and aesthetic features.

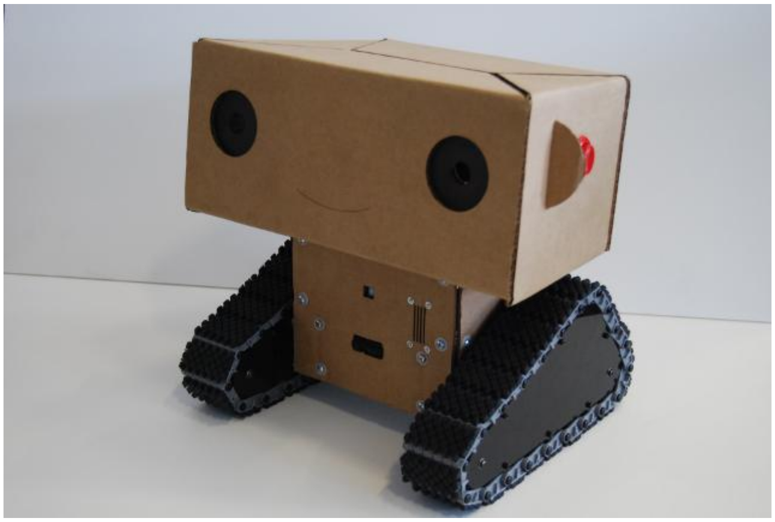


Figure 3.10: Final robot design, fully assembled.

The way the user interacted with the robot was by picking it up and pressing buttons. It was determined that putting the buttons on the side of the head would lead to less false positives from robot movement pushing buttons. Also, with the buttons on the side rather than the top of the head, a user would be less likely to randomly press a button without interacting with the robot. Multiple configurations were tested to determine the best way to pick up the robot. The original intent was to put a handle on top of the robot's head. However, in testing it was determined that placing a handle in that position encouraged to uses to hold the robot down at their side. This would put the robot in more danger of striking an object or smashing into the user's hip. Therefore, it was decided to make the bottom part of the robot's head graspable. This allowed the user to pick up the robot by the head. This was a natural instinct for users, who either picked up the robot by the head or by the tracks.

3.2 Capturing Stories

3.2.1 Behavior

Creating behavior which was "cute" was not as immediately apparent as designing the physical appearance. The behavior of the robot needed to be scripted, just as an actor would be. The particular behavior of the robot depended on its motive and its current situation. Depending on the goal of the system, the robot modified its behavior to best achieve the objective. The voice and movement of the robot were the main avenues to represent behaviors.

Through movement, we could play with the personality of the robot dynamically. Non-verbal behavior can express as much as and augment the content of verbal communication [17]. Movement is an important factor for displaying expression in robot with constrained appearances such as Boxie [18]. The nominal movement behavior of the robot was to seek out people while getting stuck or lost. The robot was made to act naive by appearing to not know where it was going. By appearing lost, people around the robot would be compelled to help it, just as they would with a lost child or helpless animal. When the robot got itself suck, its behavior was reminiscent of a wounded animal or a child in need. For example, if the robot wedged itself in a corner or under an obstacle, it squirmed around back and forth as if it is trapped. The robot would detect if it was stuck by using its distance sensor and accelerometer. When the robot got itself suck, it would raise slightly off the ground, which could be detected by an increase in acceleration due to gravity and an increase in the robot to floor distance. It would wait to detect reflected body heat or time-out in software and move again. When the robot came to a complicated intersection, it moved back and forth as if it was searching or confused about what it should do. This confusion led people to believe the robot was lost or helpless to find its way. The robot slowed down before this behavior to mimic hesitance or fear of the path ahead. When the robot sensed a human it stopped in the direction of the human and motioned toward them, indicating that it would like to initiate a conversation. This indication of conversational intent was an important factor in the capture of a human for interaction [19].

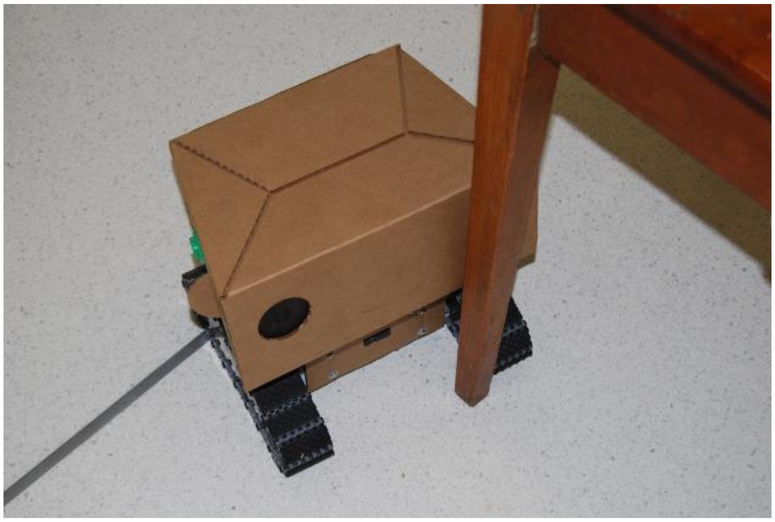


Figure 3.11: Example of the robot getting itself stuck on an obstacle.

3.2.2 Scripting

The voice of the robot was of major concern, as it was the major output medium of the robot. The success of the robot was contingent on the voice being clear and cute. The script for the voice also included elements of naive thought and minimal knowledge. Eckman et al. found that as voice pitch increased, the believability and trustworthiness of the speaker increased [20]. It was also found that the intonation of the speech carried a significant amount of information about the motivation of the agent speaking [17]. The voice actor we used for the robot was able to produce a high pitched childish female voice. This combination produced the most optimal configuration for leveraging humans and subjectively matched the appearance and behavior of the robot.

The type of story we chose to capture was the documentary. We crafted a script to capture a story about the current place the robot was in. The voice of the robot was scripted in such a way as to complete the robot's goals. Two scripts were produced and implemented. One involved giving the user simple commands while the other added personality and provided a two way conversation. The intent was to study the difference in user responses in comparing the types of scripts implemented. In “Relational agents: a model and implementation of building user trust” it was found that adding “small talk” to the interaction paradigm of an embodied agent increased the user trust for that agent [21].

At the beginning and end of each script, the robot would make an edit point in the internal camera. This separated the individual interactions inside the robot, effectively pre-editing the footage. The start and stop of a segment is the most basic form of editing. The first script we will discuss is the command based script (each new line represents a pause in the voice actor’s speech):

I need help
If you can help me
Press the green button on the side of my head

This was the first thing a user would hear when interacting with the robot. This message played when the passive infrared sensor detected body heat, or when a button was pressed. This phrasing was important because it had to be short and had to draw the user in. The first line “I need help” implied to the user that the robot needed their assistance. By asking for help, the robot made it appear that it needed a user and was not simply an autonomous device. The second and third lines made it clear that the user was empowered to help the robot, and by pressing the green button they acknowledged that they can do something help the robot. It also provided a signal to the system that the person whom the PIR detected may be willing to interact. This phrase also introduced the robot’s language, using “me” and “I”. The purpose of this was to anthropomorphize the interaction, making it seem like the robot was self aware.

Can you pick me up?

This question was more important than it may first appear. On a practical note, the robot was short and needed to be brought up to eye level in order to conduct an interview. More importantly, picking up the robot was a deliberate action which the user must perform that shows the system sustained interaction. The act of picking up the robot took more effort than pressing a button, hence represented a greater level of user-robot interaction. It also created more of a physical connection with the user. The robot used its sensors to detect when the user picked up the robot and continued its interaction after that point.

Can you put me on a table?

The robot needed to be put somewhere where it could interview the user. The sensors on the robot detected when the robot was placed on a surface and continued its interaction after that point.

Please make sure your eyes are level with my eyes so I can see you.

Since the camera was mounted in the robot's eye, we needed to make sure the users eyes were level with the robot's so it could get a good view. It also made it seem like the robot was able to see the user, further anthropomorphizing the interaction.

Please step back a little, so I can see you better.

We further positioned the user for optimal capture, if the robot sensed the user was too close, it told them to move back. The robot repeated this phrase until the user moved further than 0.8 meters from the robot (detected by an ultrasonic distance sensor). When the user moved to the correct distance, the interaction continued.

I am going to ask you some questions

When you are finished speaking

Press the green button

If you would like to hear me again

Press the red button

This blurb prepped the user for the interview section of the interaction. It informed the user that it would be asking them questions and instructed them how to interact with the robot. It also made it clear the robot could repeat questions on the user's request. It was found that users sometimes needed to be prompted again to complete the interaction properly.

How are you?

What's your name?

Where are you from?

These questions were asked to establish an initial interaction with the user and to make sure they understood the red and greed button input paradigm. Asking the user how they were was a normal way to start a conversation. We also wanted to capture their name and where they were from so we could potentially ask the user questions at a later date.

Why are you here?

This was a very broad question which could be responded to in several ways. It could be interpreted as "why are you in the area you are in now" or as in something such as "why are you in this state". Keeping the question broad allowed us to see what type of information the user was most likely to share.

What do you think about the lab?

Our initial tests were conducted within the Media Lab. We wanted to see what the user would tell us about the lab. We could then see how this changed based on other parameters.

Can you take me to your favorite place?

When we are there press the green button.

Since one of the system goals was to move the robot by leveraging human mobility, we integrated this action into a question. This also allowed the robot to capture more footage of the area for the documentary.

Can you tell me about this place?

This question was asked in order to elicit a story out of the user. We wanted them to either tell the robot a personal story about the area or tell it about what was currently happening.

Thank you for answering my questions

If there is someone else I can talk to

Bring me to them then press the green button

Otherwise press the red button and put me back down

This gives the user an option, either bring the robot to a friend to continue interaction or the release the robot back. Either way, the robot would roam through a new area since we asked the user to bring the robot somewhere interesting. This ends the first script.

The second script was conceived as pseudo two-way communication. We also added more personality to the robot, trying not to issue commands. We used lessons learned from the first script and incorporated them into the second.

I need help

If you can help me

Press the green button on the side of my head

Same start as the last script, kept for its succinct nature and successful testing.

Hi! My name's Boxie, I'm from the Media Lab and I'm making a movie about MIT! This movie will be featured on the Media Lab website.

On the side of my head are buttons so you can give me instructions. If you would like to be part of the movie, press the green button. If you want me to go away, press the red button.

This section differs greatly from the previous script. First, the robots name was introduced, creating a connection with the user. It also gave the users a name to call the robot during interaction if they wanted to address it. The robot then explained where it was from and what it was doing. This lowered suspicions that the robot may be a harmful agent. It also motivated the user to interact with the robot further. It then explained what it will be doing with the movie it was making, thus completely notifying the user of its intentions. It then went on to explain the button input paradigm and explained the current options to the user. The user then had a clear choice, either be in the robot's movie, or command the robot to move along. This was an important step to give the user empowerment over the robot. We discovered from the first simple script, that the users would be wary of the robot's intentions and wanted to be able to send the robot away if desired. By spelling out the robot's goals, we completely informed the users of what would happen and why the robot was there. If the user decides to continue, they move on to the next section.

Awesome! We're going to have so much fun!

I'm really short, can you put me on a table or hold me so I can see you?

We acknowledged that the user wishes to continue by assuring them that continued interaction will be fun. This was important as it established two-way communication, a direct response to the button press indicating a continuing interaction. We implemented "small talk" for an informal conversation… that is, that the robot would use predictable responses that the user might expect from the conversation. These predictable "thanks" and "cool" responses are common in trustworthy human conversation [22-23]. We then asked the user to pick up the robot. We learned from the previous script that some users would give up if they could not specifically find a "table", even with other places such as bookshelves to place the robot upon. Therefore, we expanded the options to placing the robot on a table or holding it. We also informed the user why they were picking the robot up. The robot self-described itself as short and told the user it needed to see them, so if the user could not find a table or hold the robot they would understand the intent of the instruction. Indeed, in testing, some people opted to squat, kneel or lay down with the robot in order to be on its level.

That's better, thanks! Can you make sure your eyes are level with mine?

I need to see you well so I can put you in my movie.

We then thanked the user for their effort and made it seem like it was a relief to the robot. The robot then asked the user to position their eyes level with the camera. We then explained that this was to see them properly for the movie. Again, making it very clear what the intention of this action was and providing a two-way interaction,

You're too close, it's hard to see you and it's freaking me out.

Can you move back a bit please?

If the sensor detected the user as too close to the camera, we asked them to move back. Here we used humor as a method to encourage compliance. After the user moves to the specified distance, the interaction continues. It was ensured at this point that the robot had a clear view of the person.

I am going to start asking you some questions to put in my movie.

Let me know when you are done talking by pressing the green button on the side of my head.

If you don't understand me, you can press the red button and I will say it again.

clears throat

What's your name?

This is where the interviewing session began. The robot reminded the user that the questions they were about to be asked would be in the robot's movie. The robot also reminded the user of the interaction paradigm involving the button input. The tests with the first script showed that there was confusion if the instructions were only given once as a command. The conversational method employed here reinforced the instructions given earlier. The clearing of the throat sound added a bit of comic relief to the serious series of instructions. It also served as a mechanism to break up the question from the instructions. The robot then proceeded to ask the user their name, waiting for the green button to be pressed.

I'm trying to see what everyone does around here, can you tell me what you do here?

This questions shows a sharp contrast from the last set's "Why are you here?" question. The robot first stated why it is about to ask the proceeding question, then asks it. This gave the user some background as to why what they are about to answer is important. It also motivated the user to give more in-depth answers. The question was made more specific, asking "what you do here" rather than "why you are here". This also is a strategy to attempt to get the user to provide an in-depth answer to the question.

Neat! What other things would you like to tell people about yourself?

The robot answers the last question with a "Neat!" This showed the user that the robot was listening and that their answers were important. This question gave the user an opportunity to tell the robot broad and unspecified thing about themselves.

That's really cool! I'd like to take a look around, can you take me somewhere interesting?

Press the green button when we are there so I know to start paying attention, you can tell me about things on the way.

The robot showed its appreciation for the last response before asking the user to perform another action. This question differs from the "take me to your favorite place" as it was a little more broad, asking the user to take it "somewhere interesting". This did not assume the user had a favorite place, but rather could find somewhere "interesting". The robot also reminded the user to press the green button so it could "start paying attention". This was to inform the user that they need to remember to press the button or the robot may ignore them. It also informed the user that they could talk along the way, thereby reminding them that the robot would still be recording.

Wow, good choice, this place is neat, I'd love to put it in my movie!

What goes on here? Can you show me around?

Press the green button to let me know when we're done walking around.

We rewarded the user for bringing the robot to somewhere new and reminded the user that it would be in the robot's movie. The robot then asked to be shown the area and for the user to describe what was going in there, acting like a reporter. The questions also imply that the user should hold the robot and move it around. Another reminder about the green button was given.

That was all right, but I think my movie needs something really special. Do you know any cool dance moves?

I'll put on some music so you can rock out.

Move far back from me so I don't get hit by your awesome moves.

I'm only going to play it for 30 seconds, so dance fast!

(pause)

Push the green button to continue.

It was decided that the robot should try to get the user to do something that they would not ordinarily do. If they complete this action, we would know that they had completely followed the robot's goals. It was found that dancing for the robot would fulfill these requirements. Dancing was something that would not ordinarily be done in the places the robot traveled. Dancing was also a personal activity that has the possible outcome of being embarrassing. Dancing also made for interesting video footage. First we told the user that what they had done before was ok, but what they were about to do next would be really special. We asked the user to make sure they are far away from the robot, so they don't hit it and so they are far enough away from the camera to see the user's entire body. We also informed the user that they would only have to dance for 30 seconds.

Here we go!

(pause)

music

The robot waits for a bit, encourages the user, waits some more then begins playing music.

Wow, that was AWESOME! You've been great, thanks so much for helping me out with my movie! Can you tell me your e-mail address so I can keep you updated about the movie?

We make sure that we thanked the person for their hard dance work. The robot then asked the user for an e-mail address for possible contact later.

Thanks, You can set me down and press the red button so I can roll around and find more interesting things to film, or you can give me to a friend and press the green button so I can talk with them. Thanks again, you're the best!

This is much like the end of the first script, asking the user to give the robot to a friend or put it down. We thank the user and end the interaction.

Chapter 4

System Design

4.1 Mechanical Design

4.1.1 Overall Mechanical Considerations

The mechanical design and interaction design were closely tied. The success of the robot's interaction was contingent on a well planned and executed mechanical design. The weight of the robot was constrained by the interaction, since we would be asking users to pick up and carry the robot. Lower weight also allowed the motors to work more efficiently, which saved power consumption and allowed to robot to rove for longer periods of time. We based our weight goal on the weight of a heavy laptop, approximately 7 pounds. To achieve this, we considered the heaviest parts of the robot, those parts being the chassis, casing and battery. In the design of the chassis and shell, we focused on finding the lightest and most robust material that would still be aesthetically pleasing. Size was also considered an important factor. We did not want it to be so small that it would lose itself under materials in the environment and we did not want it so big that carrying it around would be unwieldy. The size of a large cat was chosen as one that would fit with these requirements.

Users were anxious to pick up a robot that looks fragile. If they are afraid that their interaction will harm the robot, they are less likely to carry out those interactions. Furthermore, since the robot was released unattended, there was a high possibility that the robot would be in situations where it would be put under stress. Some of these situations included being dropped after being picked up or having something fall on the robot. Because of this, robustness and reparability of components was taken under consideration. If a user interacts with a device and finds it to be poorly made or not robust, they lose confidence in the system and are less likely to continue interaction.

4.1.2 Chassis

The chassis includes the treads and base plate that supports the robot. We decided to use treads in lieu of wheels because they are more robust, easily climb over obstacles and provide a flat surface for a user to hold the robot by. The treads, motors, sprockets, idlers and tread panels were acquired from Lynxmotion, a company specializing in components for small robots. The treads were made from interlinking pads and had a width of 2 inches and a pitch of 1.07 inches. The pads were molded from polypropylene and rubber which had a Shore A hardness of 45, providing a very wear resistant surface. Twenty one pads link together using twenty one Delrin rods which slide between the interlocking hinges of the pads to create a pivot point. Since Delrin is a self lubricating polymer, we did not have to worry about excess friction or greasing the joints. Each rod was held into place by two black nylon snap rivets (see Figure 4.1). Fully assembled, each track had a linear length of approximately 23 inches (with slack from the hinge points).

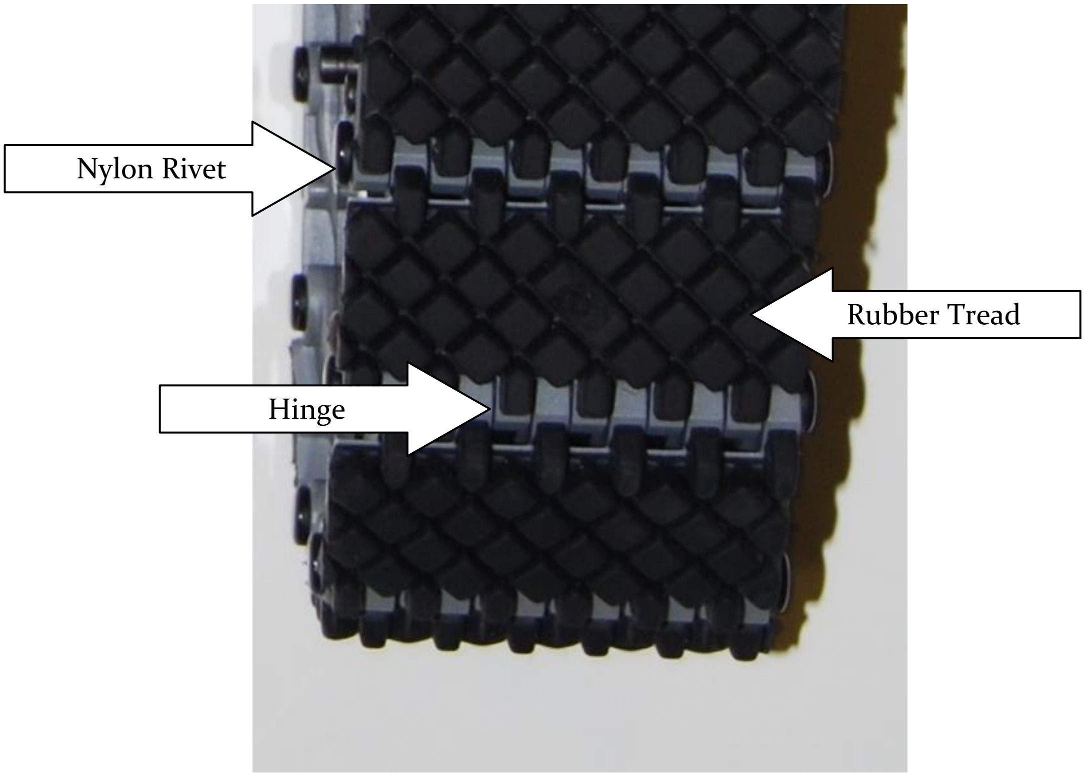


Figure 4.1: Close-up of assembled track

The drive system was completely contained within each tread mechanism. The robot moved by the skid steer method. Skid steer uses the differential speed of each tread to steer, i.e. if the robot wanted to turn left it would slow down the left tread to "skid" that tread and turn left. Since the robot used skid steer, there were no complex steering mechanisms required. This kept weight and possible points of failure to a minimum. The tread was wrapped around three sprockets and six nylon idlers. The top sprocket in the triangular tread system was the drive sprocket, which was directly coupled to the gear motor. The other two sprockets acted as idlers; they were connected to the chassis with roller ball bearings. These idlers had teeth to engage with the tread and keep it tracking properly. Six nylon idlers provided extra support, however these did not engage the tread. There were six hex standoffs which coupled together the two laser cut acrylic side panels. These side panels were what held the mechanism together and provided mounting points for the base plate. The bearings from the sprocket idlers and the hex standoffs went to the side panel and the hex standoffs were secured with hex head screws. The hex standoffs provided the support for the panels which in turn provide support for the components. Figure 4.2 illustrates the drive system with the front plate removed.

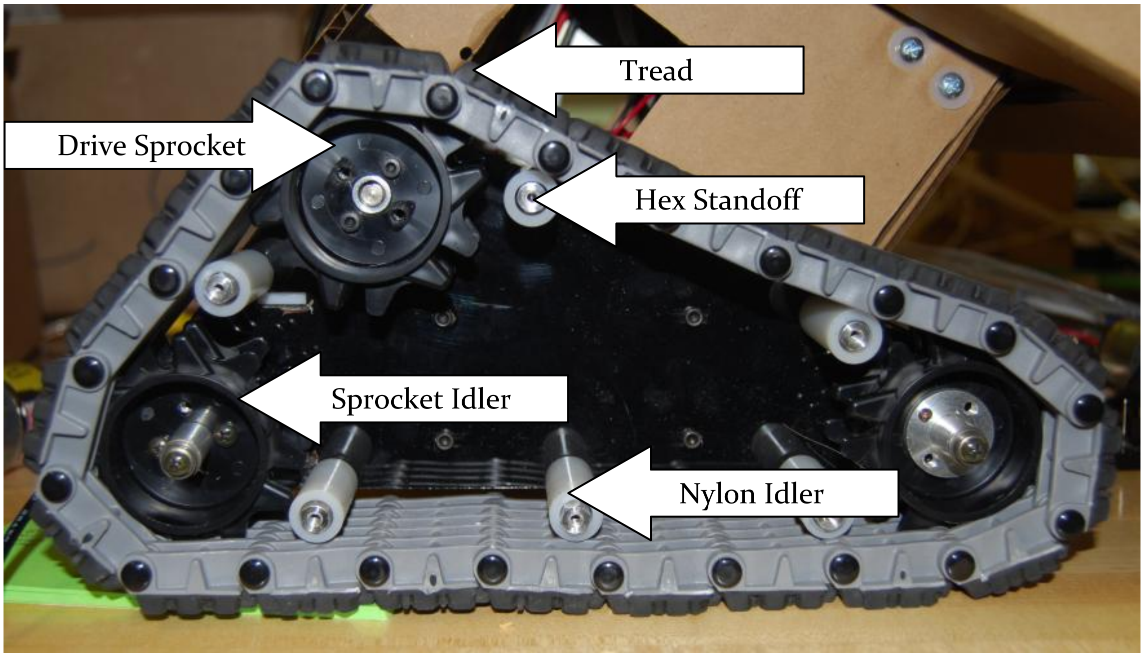


Figure 4.2: Side view of tread drive system internals, showing individual components (note: front plate is removed in order to see internals).

The base plate provided the structural support for the robot. The mechanical stress of the robot in operation was transmitted to this plate. The plate consisted of two laser cut acrylic panels. The panels were designed in SolidWorks and converted to 2D CAD drawings (see Figure 4.3). The panels were fastened to the treads by way of rectangular aluminum bars with holes tapped in them. The top panel was then connected to the bottom panel by way of hex standoffs. Each base plate also had right angle clips mounted on their periphery. These clips allowed the shell to be mounted to the chassis. This interconnection of components made for a stiff and sturdy base for the rest of the robot to be built off of. Figure 4.4 shows a close-up view of the attachment points. Figure 4.5 shows a top view of a fully assembled chassis (less the electronics).

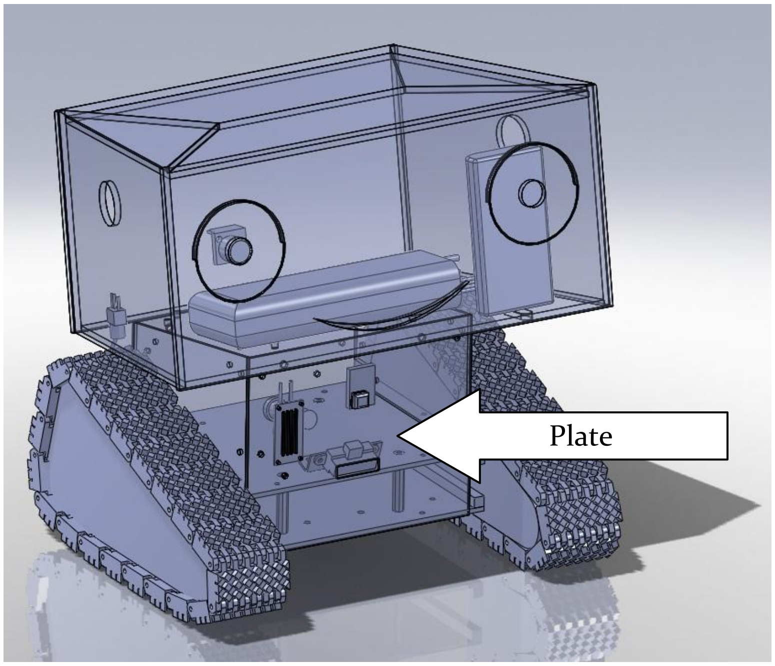


Figure 4.3: SolidWorks drawing of plate, later converted into a 2D rendering.

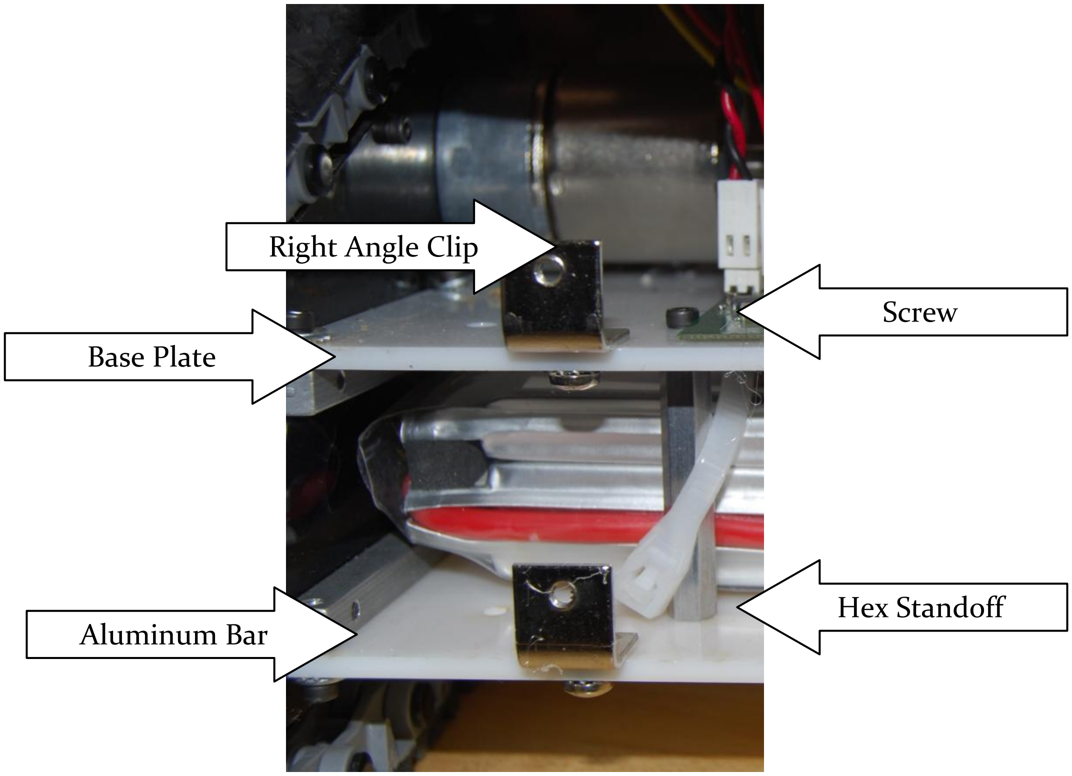


Figure 4.4: Rear close-up view of base plate attachment points.

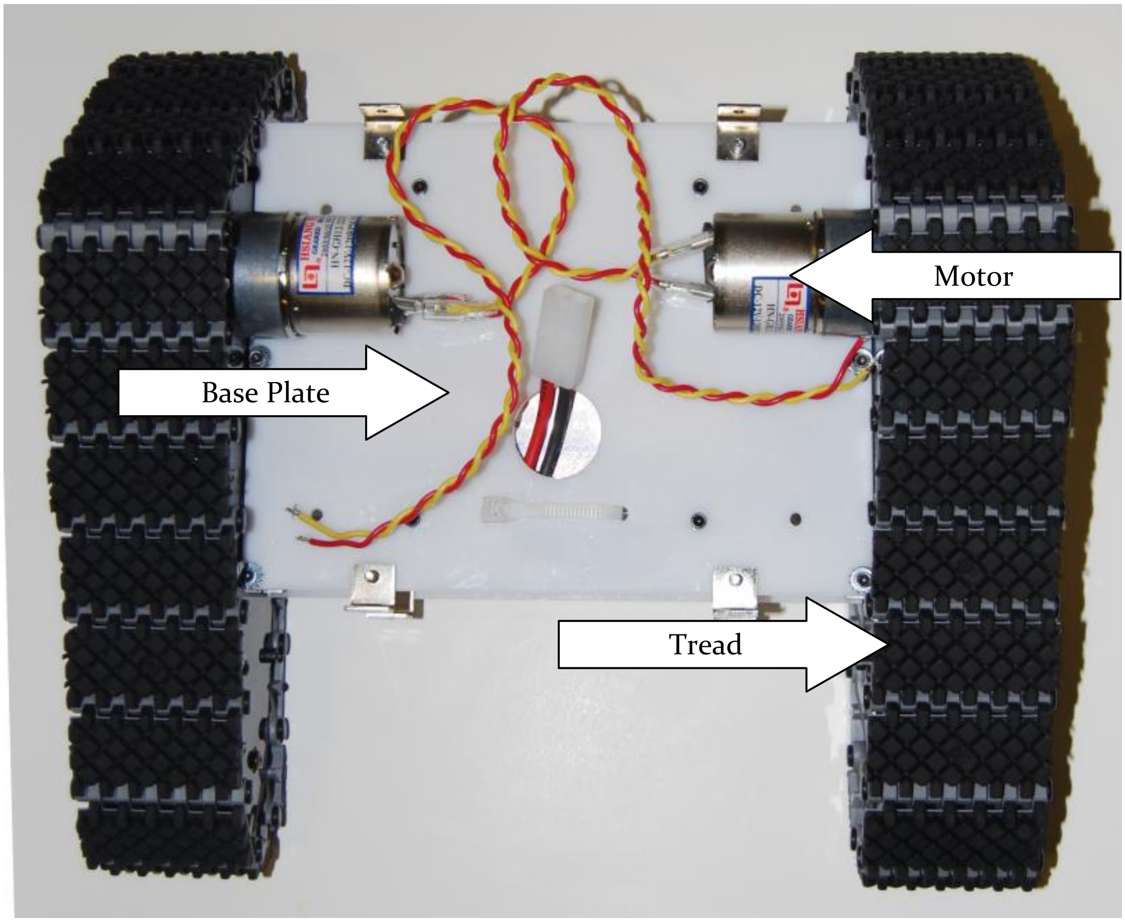


Figure 4.5: Top view of robot treads and base plate showing fully assembled chassis.

The base plate also housed the accelerometer, power regulator, audio amplifier and battery. The accelerometer was mounted on the base plate because of its rigidity and its close coupling to the drive system. The battery was placed here to lower the center of gravity of the robot which made it more stable. Figure 4.6 shows the components mounted on the base plate.

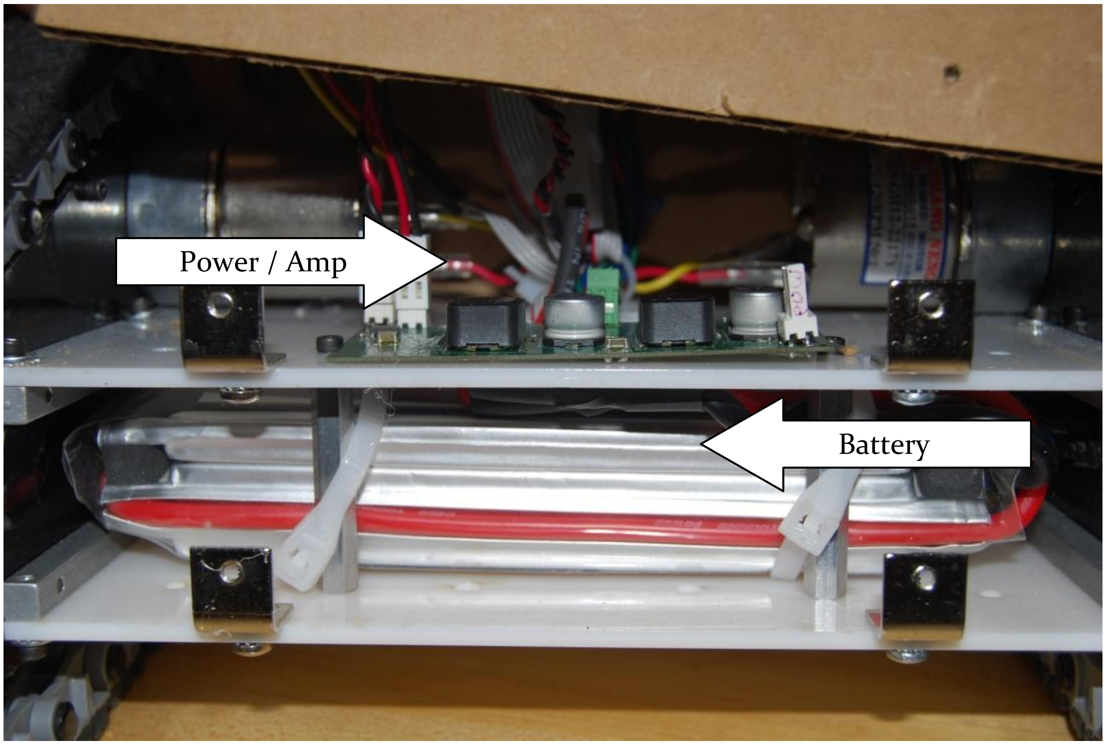


Figure 4.6: Battery, power regulator and audio amp mounted on the base plate. The accelerometer is located on the opposite side (not pictured).

4.1.3 Shell

The construction of the shell of the robot was one of constant evolution. This was due to the fact that the mechanical design of the shell and the interaction design of the robot were closely tied (see section 3.1.3 for design considerations). However, since the minimal size of the head was contingent upon the components that needed to be put inside, the first step was to determine the minimal dimensions. The minimal dimension was determined by the largest component in the head, the camera. The dimensions of the camera were 3.94 inches high by 1.97 wide by 0.66 inches deep. Since the lens of the camera needed to be in the center of the robot's eye, we were able to determine the position of and model the camera into a head which fit both our design specifications and the specifications of the camera (see Figure 4.7).

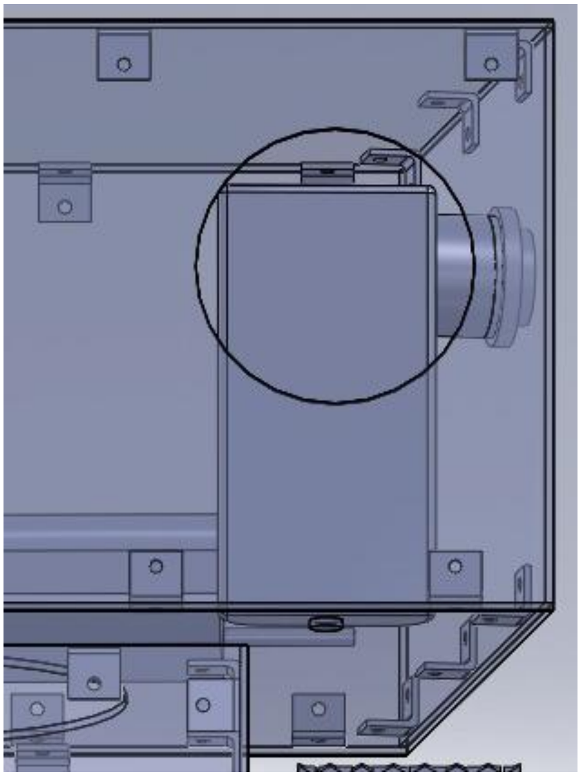


Figure 4.7: Transparent model close-up of camera mounted behind eye socket. The height of the camera informed the minimal height of the head.

After having this rough head design, we considered the other components which would have to be mounted in the shell. The components in the shell included two push buttons, the control panel, an ultrasonic distance sensor, a passive infrared sensor, an infrared distance sensor, and a speaker.

The head housed the camera, control plate, ultrasonic distance sensor, and push buttons. The control panel had the microprocessor, mp3 player, wifi transceiver and power bus (including a fuse and main power cutoff) on it. It was decided that the buttons would be put on the sides of the head. Each button (one red and one green) was put on one side behind a superficial ear. The ears added another element of anthropomorphism and also protected the buttons from accidental presses. Holes for the buttons were laser cut along with the panels and the buttons were held in by a retaining ring from the inside. In this way, the button tops were flush with the side of the robot. The camera was mounted by connecting a ¼”-20 screw through a slot in the base of the head into the camera’s tripod mount. This allowed the camera to be securely attached while still being adjustable. It was important that the camera was securely mounted to avoid it shifting when the robot was handled. The ultrasonic sensor needed a clear view of the terrain ahead to properly path-plan. The best place to mount the sensor, which did not interrupt the aesthetic qualities of the head, was in the center of the other eye. This was also an optimal location to measure the distance to the user during video capture. The control panel rested on the bottom of the head with wires snaking through an opening into the base of the robot. The back of the head had a hole cut in it covered with a panel and attached with Velcro. This allowed for us to gain easy access in order to download the video from the robot, update the sound files, turn the robot on and off and perform maintenance. Figure 4.8 shows the mounting positions of the major components in the head.

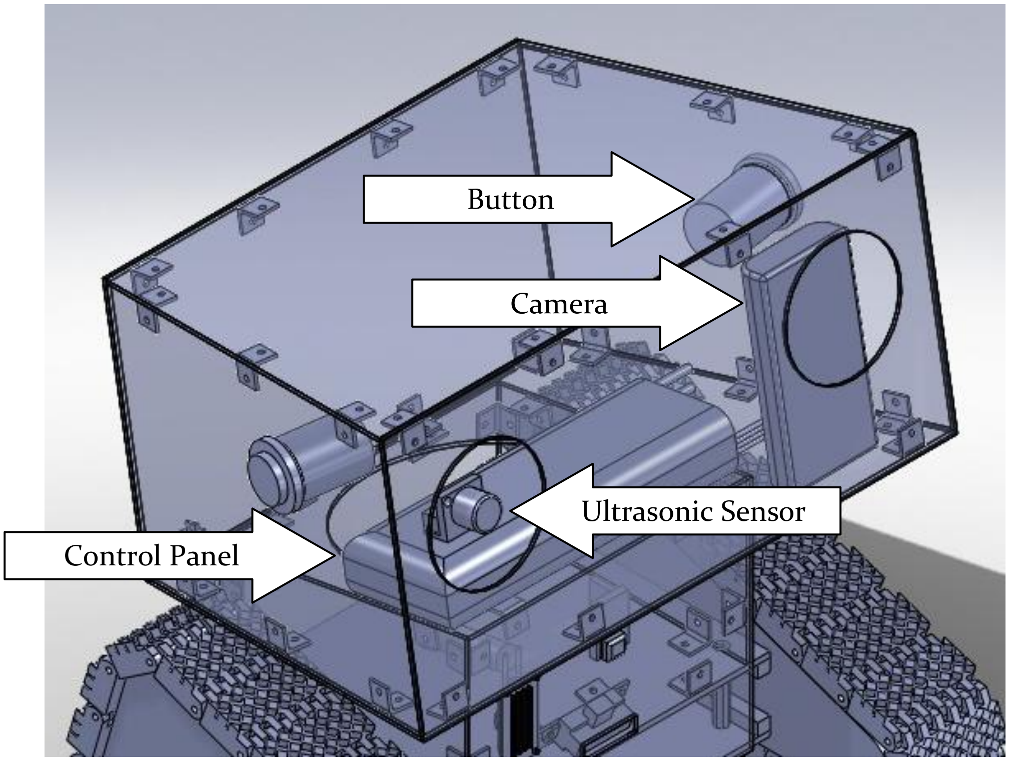


Figure 4.8: Transparent view of head showing mounting positions of major components.

On the front of the base of the shell, a plurality of devices were attached. These devices included a passive infrared sensor, hence an infrared distance sensor, and a speaker. The PIR sensor itself had a wide field of view, this would produce errors if the sensor saw something outside its designated sensing radius. To narrow down the sensing field of the device, it was mounted recessed in the base of the shell. By doing this, the head would block errant heat signals from above and the sides of the recessed hole would narrow the detection radius. It was essential to keep false body heat readings to a minimum to increase the efficiency of the person finding algorithms. The mounting of the IR distance sensor also required care. It was essential that this sensor see both a portion of what was ahead and the ground below the robot. Therefore, the sensor was mounted at such an angle that it would see approximately 4 inches past the tip of the treads (see Figure 4.9). This was done by mounting the sensor and testing the detection range while making the appropriate adjustments.

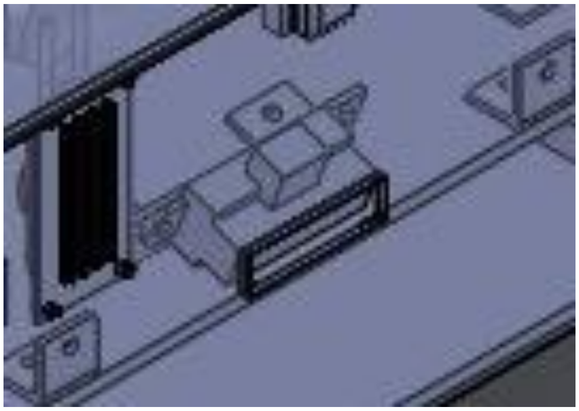


Figure 4.9: IR distance sensor shown tilted.

The compact oval dynamic speaker needed to be mounted inside the shell for protection, however since the sound needed to be projected forward, a grille was cut in the shell. The speaker was mounted firmly with four screws, which was needed in order to get the greatest sound amplitude from the resonance of the box. The speaker had to be mounted facing forward to make it appear as if the sound of the robot was coming from the side that had its mouth. Figure 4.10 shows the major components in the base of the shell.

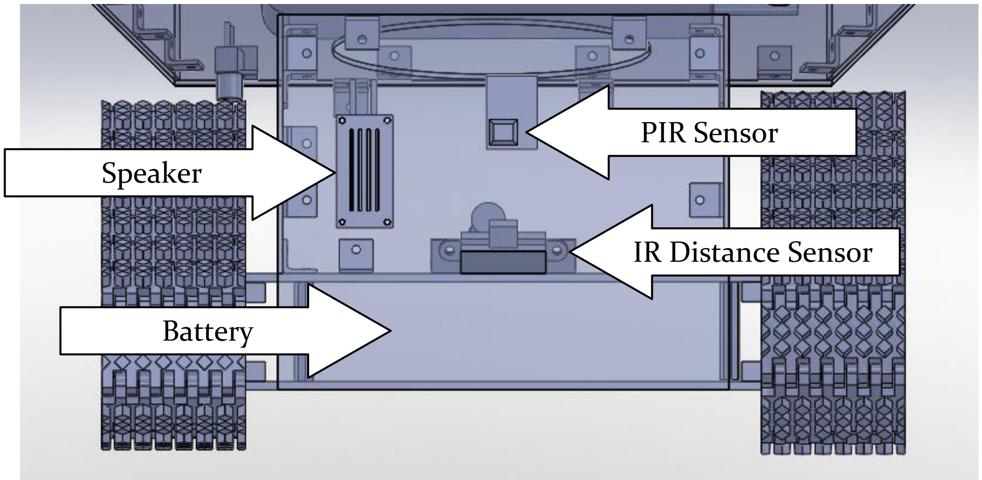


Figure 4.10: Transparent front view of robot base showing major components.

After choosing the final shell appearance in the interaction design phase and knowing where all the internal devices would be mounted, prototypes were produced. The intended final shell was made out of translucent white acrylic. The laser cut acrylic panels were held together by right angle clips and hex head screws (see Figure 4.11). However, as discussed in section 3.1.3, this design was found to not be pleasing. The material did not look "friendly" and the screws connected to the clips around the material were not aesthetically pleasing.

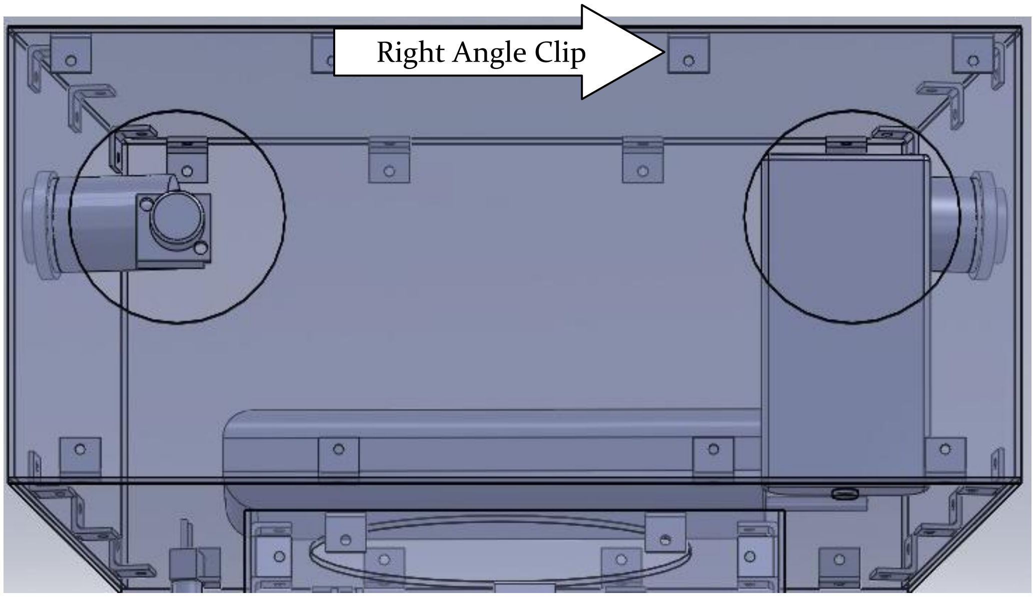


Figure 4.11: Transparent view of head showing right angle clip positions.

It was then realized that the cardboard construction we had been prototyping with was very pleasing and had a familiar and friendly feel to it. At that point it was decided to investigate techniques to fabricate a robot shell out of cardboard.

The first and most obvious place to look for robust construction with cardboard was in box design. Corrugated cardboard boxes are surprisingly robust. A quick survey of boxes around the lab showed that small boxes could hold in excess of 60 pounds and had a crush strength in excess of 100 pounds. Most boxes are assembled by cutting slots out of cardboard creating flaps which get glued back onto the box (see Figure 2.1).

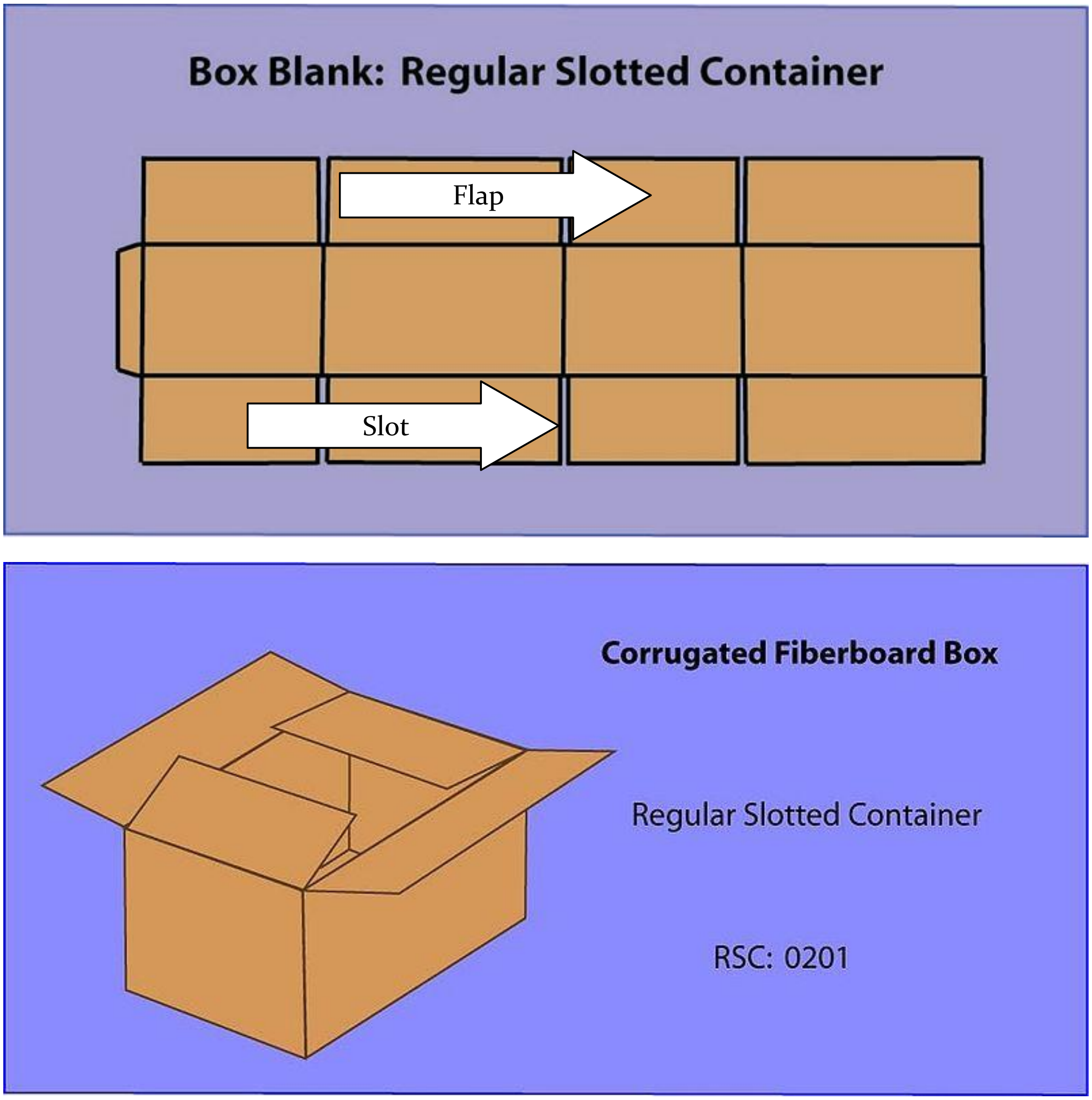

Box Blank: Regular Slotted Container
Flap
Slot
Corrugated Fiberboard Box
Regular Slotted Container
RSC: 0201

Figure 4.12: Typical cardboard box construction [24].

To convert our previous prototypes into sturdy boxes, tabs were extended from the perimeter plates. The tabs were made slightly longer than necessary to accommodate bend radius when the tab was folded over (see Figure 4.13). Each tab folded over onto a plate with no tabs and was glued there. Different adhesives were investigated including wet mount kraft packing tape and VHB pressure sensitive adhesive by 3M. After several strength tests it was concluded that the easiest and most reliable bond between the cardboard flaps and the plate was high temperature hot glue. The cardboard material would fail before the hot glue, indicating that the bond was stronger than the material. Hot glue also allowed for a few seconds of working time before the bond was permanent. This was important to allow for slight adjustments in fit.

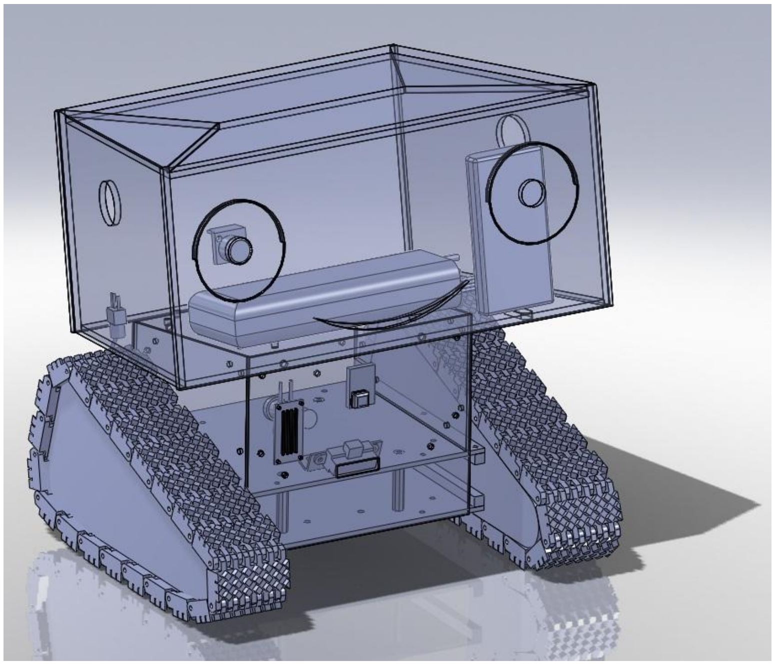


Figure 4.13: Tab design in SolidWorks.

After final CAD designs were completed, the panels were fabricated. We used cardboard sheets which were cut in a laser cutter. We chose 200#/ECT-32 corrugated, with C flute. The strength of this cardboard is high with a high edge crush test number. The C type flutes were compact enough to allow for easy and clean bending of the material. This particular cardboard cut cleanly with the laser cutter with minimal smoke. To laser cut the SolidWorks designs, we panelized them into dxf files and opened in Corel Draw. Corel Draw is the software package used to layout and print on the laser cutter. The panels were then cut and verified by dry fitting (see Figure 4.14).

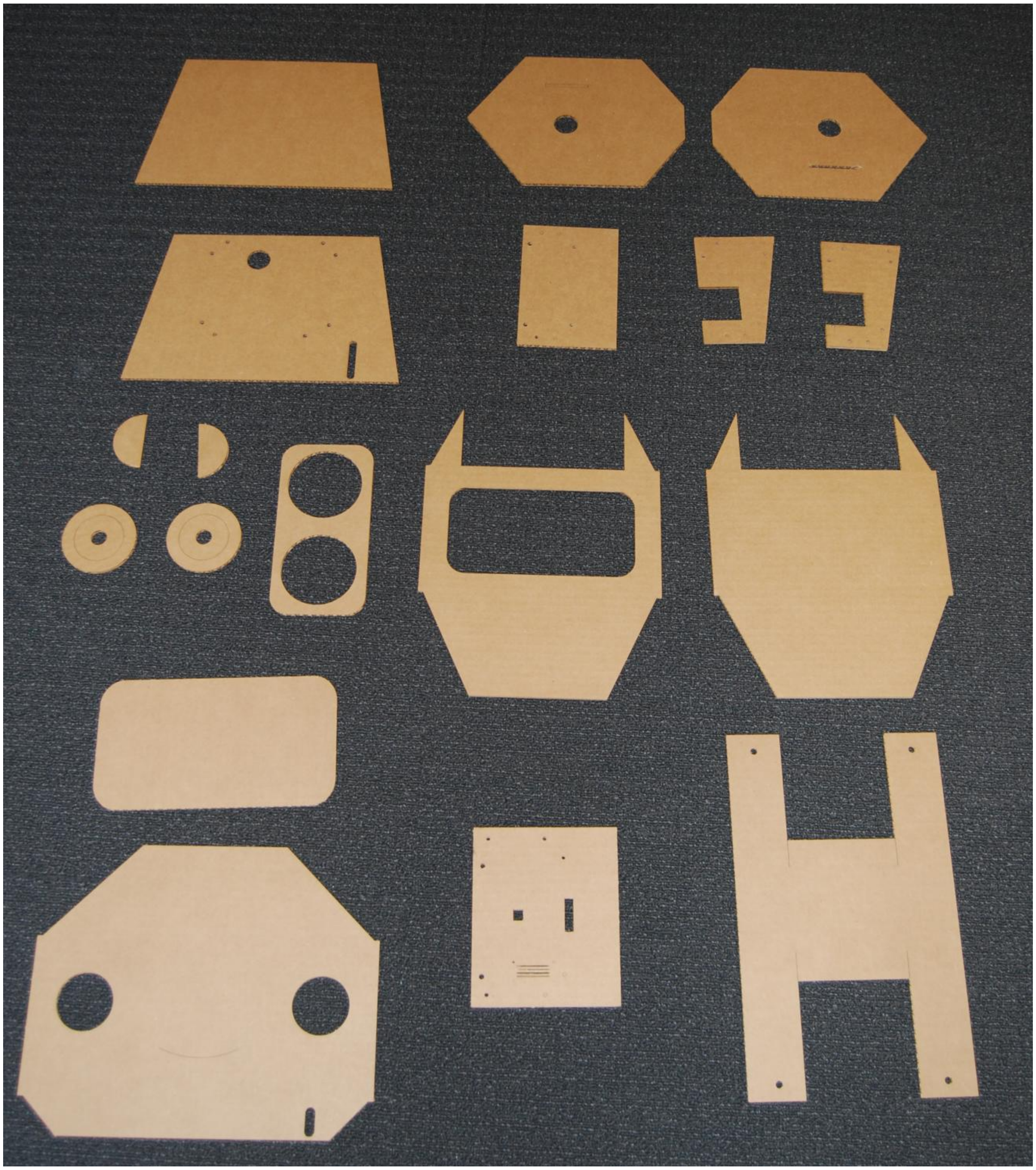


Figure 4.14: Laser cut robot pieces.

After dry fitting the components to check for fit, the panels were assembled into their modules. All the modules which required glue were assembled first. The modules that were attached with screws were assembled later. Figure 4.15 shows a fully assembled head module. Notice the tabs attached to a central plate. The design shown in this picture is nut full assembled, the tabs on the side attach to a plate beneath creating a box.

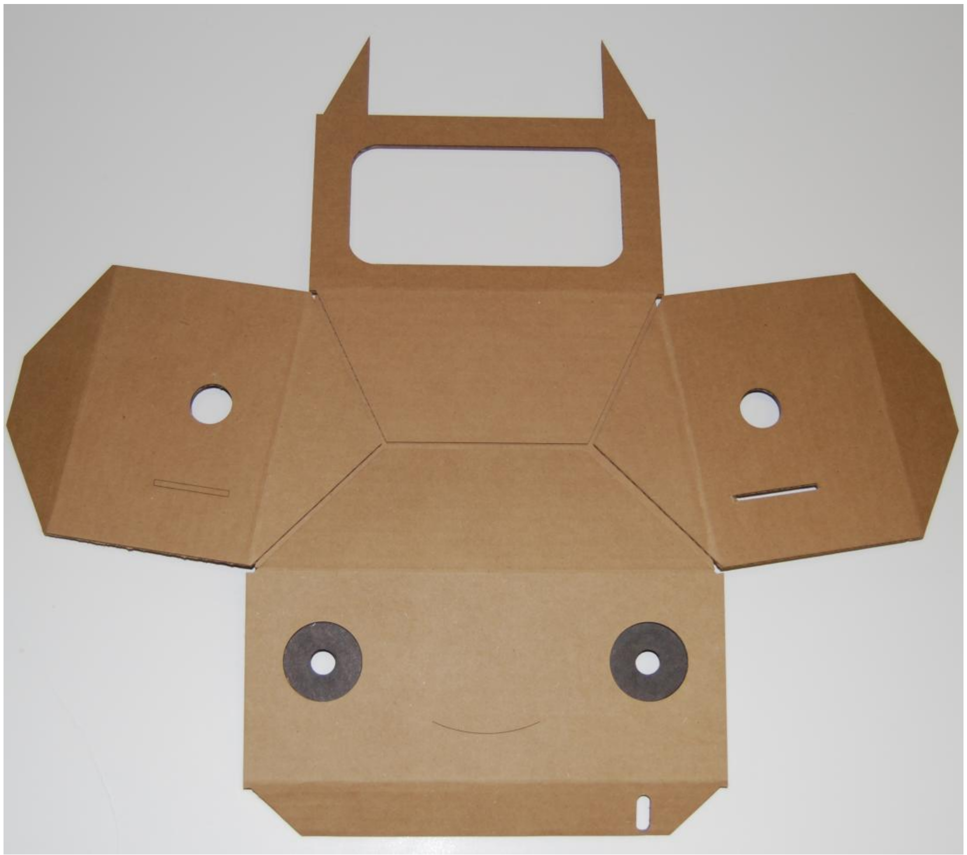


Figure 4.15: Assembled head module.

The final assembly was extremely robust - in fact it was found to be stronger than the acrylic base plate, since the cardboard was able to recover from impact. Figure 4.16 shows a piece of the base plate broken off from an impact. The right angle clip was attached to the cardboard shell. Note that the shell was unharmed while the base plate shattered.

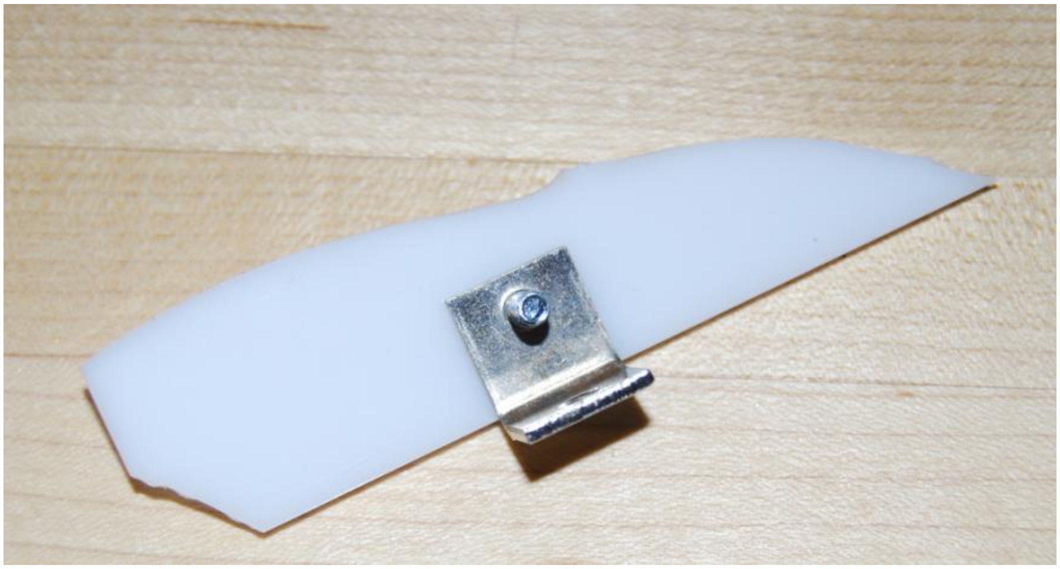


Figure 4.16: Broken attachment point from base plate.

Figure 4.17 shows the final robot mechanical assembly.

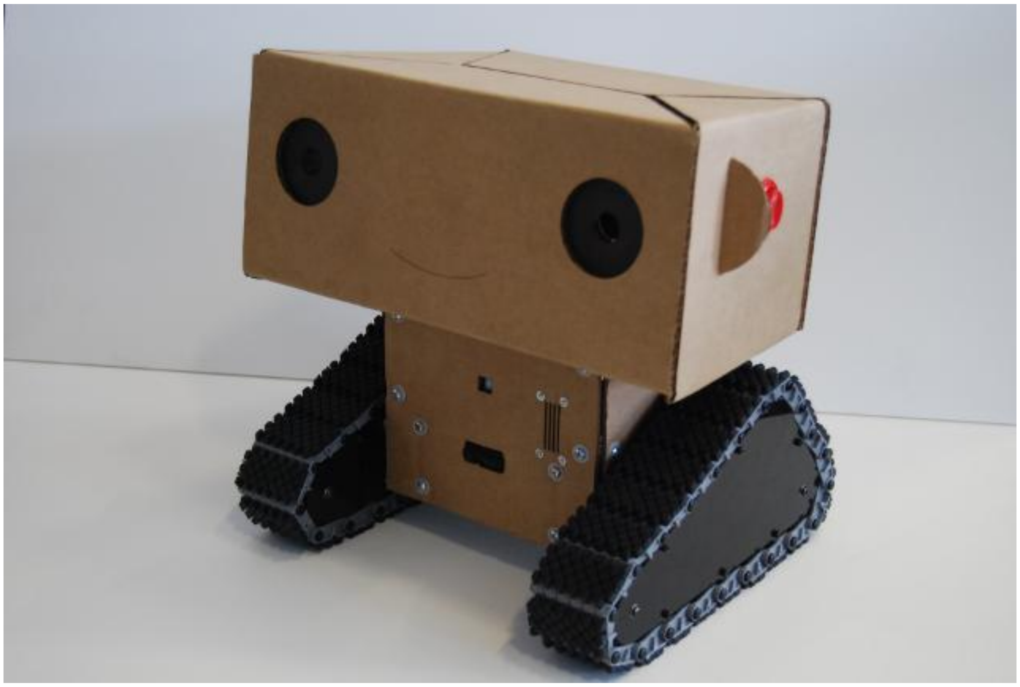


Figure 4.17: Fully assembled robot.

4.2 Electrical Design

4.2.1 Electrical System Block Diagram

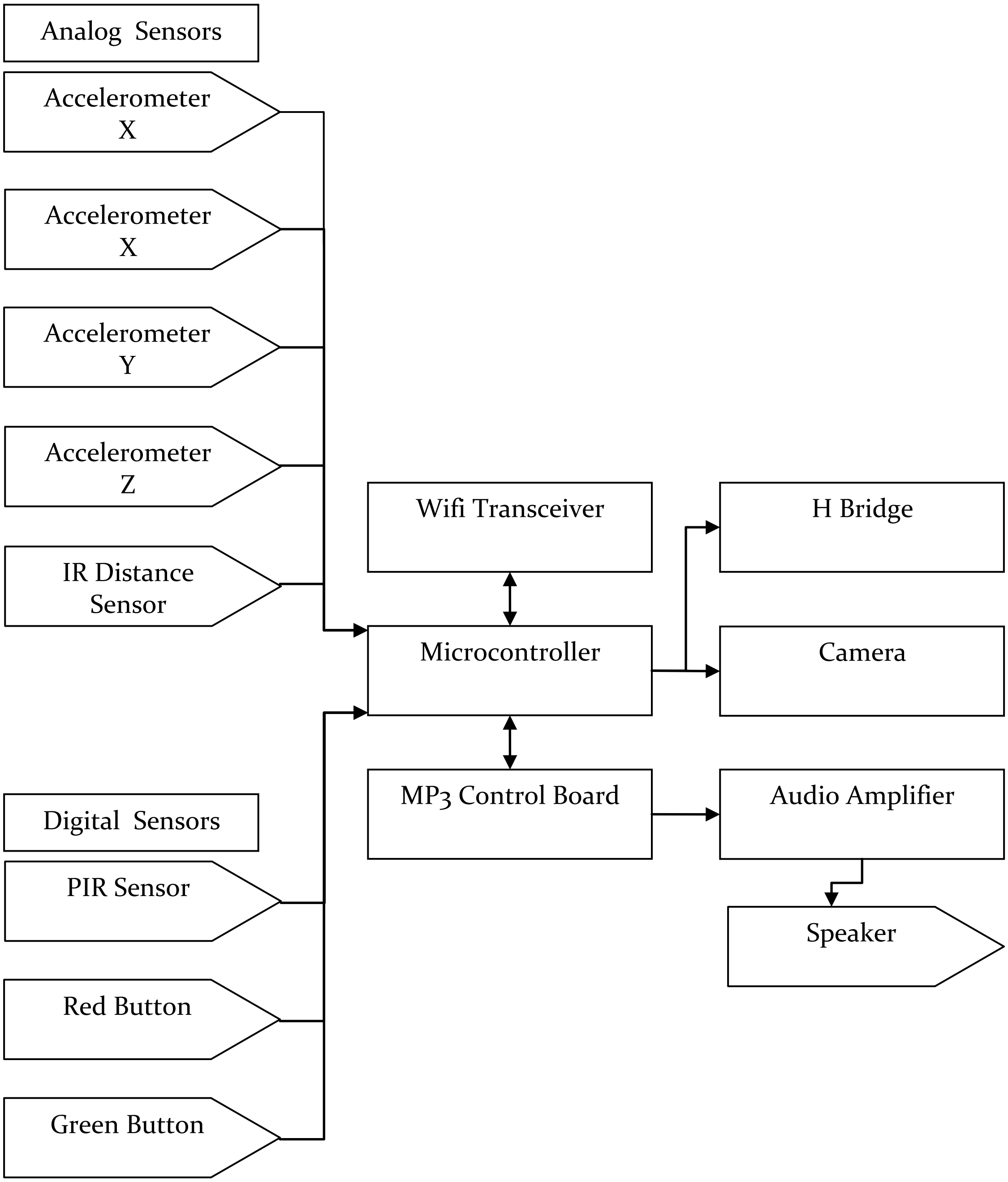


Figure 4.18: Electrical system block diagram.

4.2.2 Logic

The main microcontroller chosen for the robot was the ATMega1280 from Atmel, on the Arduino Mega development board. This board was chosen for its proven hardware design and the features available on the ATMega1280. The ATMega1280 is an 8 bit RISC architecture microcontroller, which is capable of up to 16 MIPS at 16 MHz. [25]. The Arduino Mega has many of the AVR microcontroller's pins broken out and also features an onboard USB to serial circuit and power regulation. This robust hardware allowed the development of the robot to progress quickly. The pins broken out included PWM channels, four UART modules, dozens of digital I/O and sixteen analog inputs (see Figure 4.19). Since the robot would be using a plurality of sensors and several devices requiring bi-directional serial control, the large number or UARTs and analog inputs were needed. The board also included a boot loader supporting the gcc library, which allowed us to write standard C code and easily write API libraries for the various peripherals on the robot (see section 4.3.1 for more information).

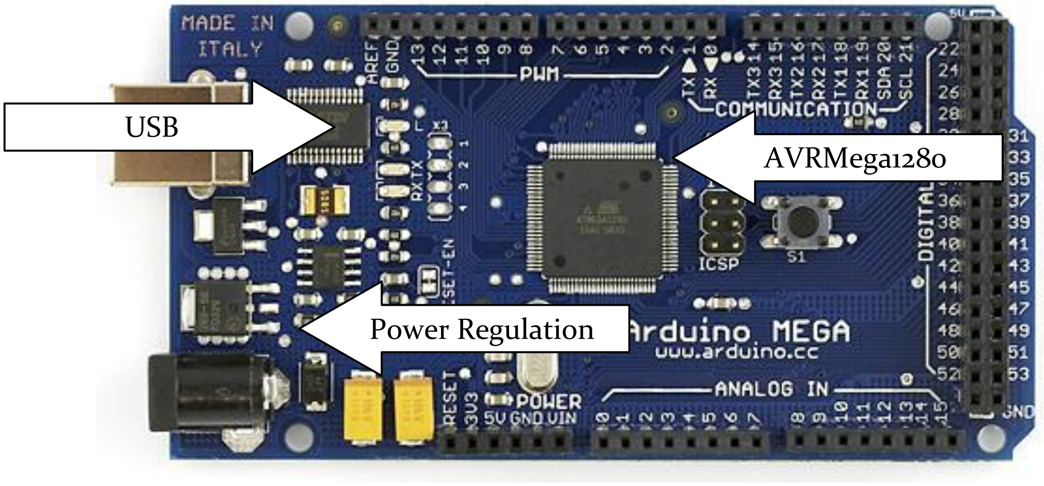


Figure 4.19: Arduino Mega Development Board

4.2.3 Sensing

The sensors for the robot were carefully chosen to be robust and low-cost. We wanted to be able to sense all that was needed to acquire and engage with a human in a minimal way. The sensors chosen for implementation included a three axis accelerometer, infrared distance sensor, ultrasonic distance sensor and a passive infrared sensor.

The robot needed to sense things such as obstacles, being picked up and rough handling. This was done in two ways; one of the methods was leveraging internal sensing.

This was accomplished with a three axis accelerometer. As low cost and reliability were concerns, the MMA7260Q3 three axis accelerometer evaluation-board from Freescale semiconductor was chosen. The sensor features a low power shut down mode and a high sensitivity output with selectable ranges (±1.5, 2, 4, and 6g) [26]. The output is radiometric based on a 3.3 volt power supply. The device was filtered through 1k resistors and 0.1uF capacitors, this filtered out the excess noise from the sensor. The sensor was also hard wired at 6g sensitivity, since the robot would need to measure full range. The sensor was firmly affixed to the base plate to get the best readings (see Figure 4.20). Figure 4.20 also shows the interconnect method employed internally. Custom connectors were made from 0.1" spaced header which mated with female headers on the various daughter boards, these connections were then glued to keep them in place. This allowed for reliable and organized internal wiring.

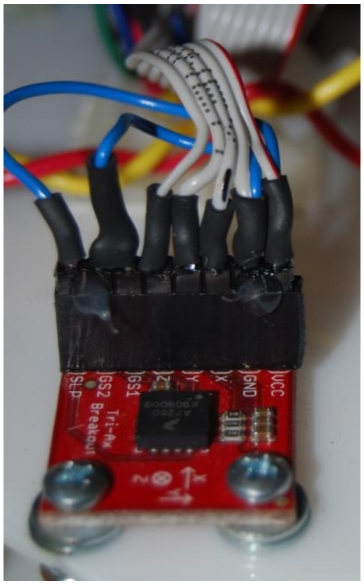


Figure 4.20: Accelerometer board mounted onto base plate.

The robot needed to know when it was approaching a drop-off point (such as a set of stairs or a table edge) and when it was being picked up. To sense these conditions, a Sharp GP2Y0A21YK infrared distance sensor was employed. The sensor uses triangulation calculations with IR light and produces an analog voltage of 3.1V at 10cm to 0.4V at 80cm [27]. The front of the sensor contains an infrared emitter and receiver which needed a clear view of the ground (see Figure 4.21). The sensor needed to be mounted at an angle which put the sensing plane about 2-3 inches in front of the tip of the treads. This would allow the robot to sense a fall-off condition with time to stop (see Figure 4.22). This angle also allowed the robot to tell when it was picked up or placed back down. This was done by determining if the robot was moving, if it was not and the distance to the floor changed, it could be inferred that the robot was picked up. If after this state the distance one again normalized, it could be inferred that the robot was put on a surface. Care had to be taken with this sensor as its input was inherently noisy and had to be filtered in software.

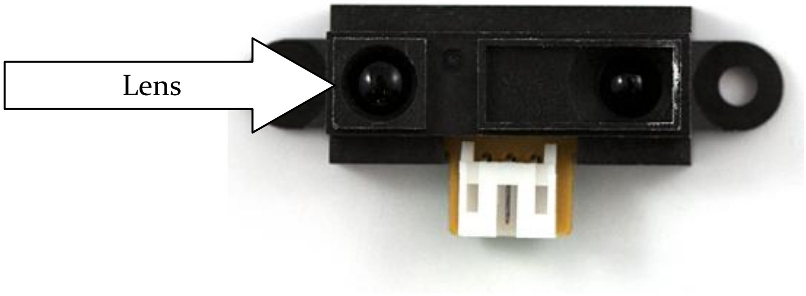


Figure 4.21: Front view of infrared distance sensor.

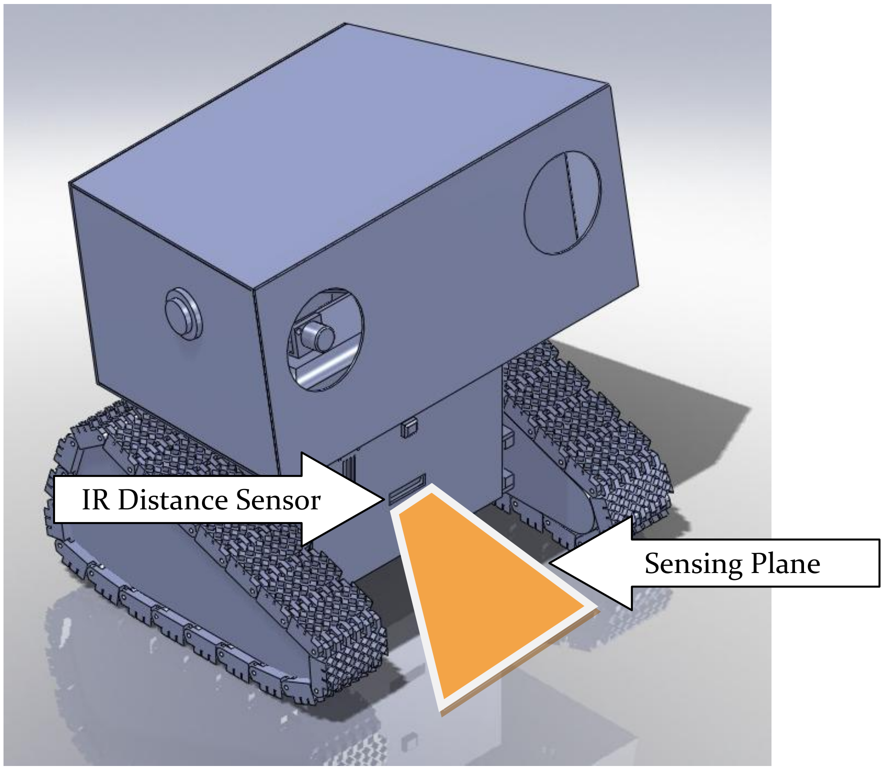


Figure 4.22: Sensing plane of the infrared distance sensor.

For overall navigation and obstacle avoidance, an ultrasonic distance sensor was used. The XL-MaxSonar-EZ0 from Maxbotics was chosen for its high stability and high acoustic output. The sensor had a 100mS (10 Hz) refresh rate and a 25 foot range with a calibrated beam angle of 30 degrees. It also featured auto calibration on the fly and support for zero range detection. The output of the device could be RS-232, PWM and analog voltage with the distance determined by (Vcc/1024) / cm [28]. This high stability and accurate ranging allowed for simple and effective navigation algorithms. The module was small with a low current draw, which allowed it to be placed in the head of the robot (see Figure 4.23). The eye was chosen as the best mounting place since it would allow the sensor to have an unobstructed view of the area in front of it (see Figure 4.24). This mounting configuration was also ideal to sense the user's distance from the camera. If the user was too close, the robot would ask them to back away to get a better shot with the camera.

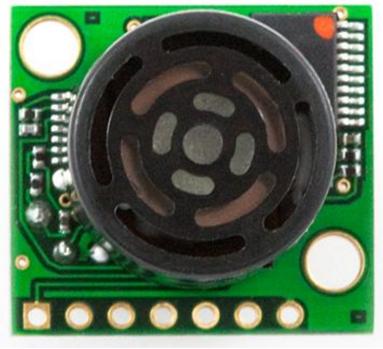


Figure 4.23: Ultrasonic rangefinder.

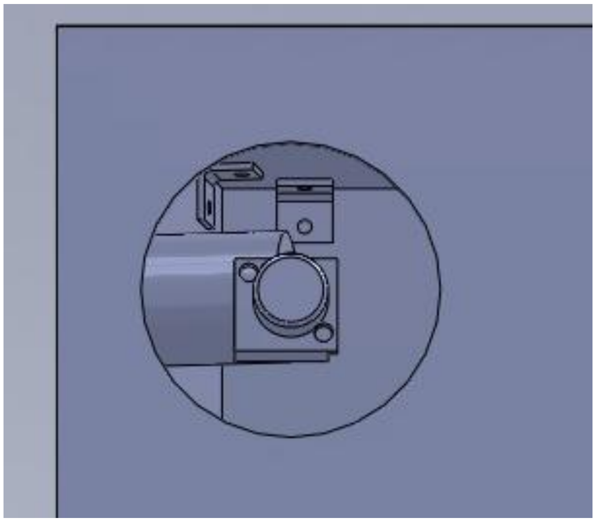


Figure 4.24: Rendering of the ultrasonic rangefinder mounted behind an eye.

The robot needed to sense the presence of people in order to interact with them. To accomplish this, a passive infrared sensor was employed to detect body heat. The sensor chosen was Zilog's ePIR, which is a motion-detecting single-board computer. The board features a Z8 Encore! XP MCU and a Murata passive infrared (PIR) sensor. The sensor is shielded by a Fresnel lens (see Figure 4.25). The Z8 MCU performs statistical processing methods in order to produce reliable person detection signals. The module has a 2.7-3.6VDC operating voltage with a typical current draw of 8.9mA. The sensor features a 5 by 5 meter 60 degree detection pattern. The dynamic PIR sensor stabilization minimizes power-on time and increases accuracy while being temperature stable. The module is configured through asynchronous rs-232 serial at 9600 baud (8-N-1) for further configuration options [29]. This module was mounted on the front of the robot, with its view angle restricted to reduce false positives.

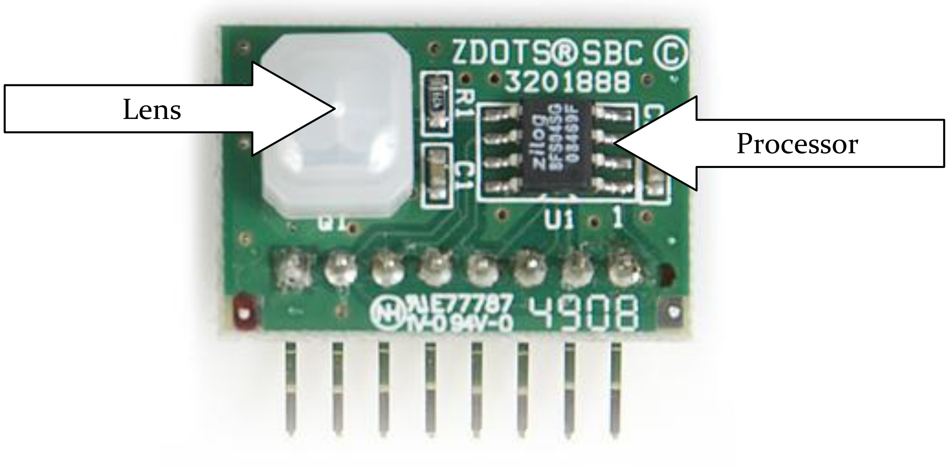


Figure 4.25: Close-up of ePIR passive infrared sensor.

4.2.4 Camera

The most important input capture device of the robot is the camera. The camera recorded both high-definition 720p video and audio of the interactions. It was also what provided the footage for the end product of the robot, the video. There are many large and expensive development kits for HD camera modules on the market. However, we took a different approach and chose to modify an existing camera module. The camera we chose was the Mino HD camera from Flip Video (owned by Cisco). The camera featured a high quality camera module, memory and battery life supporting 2 hours of video capture and integrated microphones (see Figure 4.26).

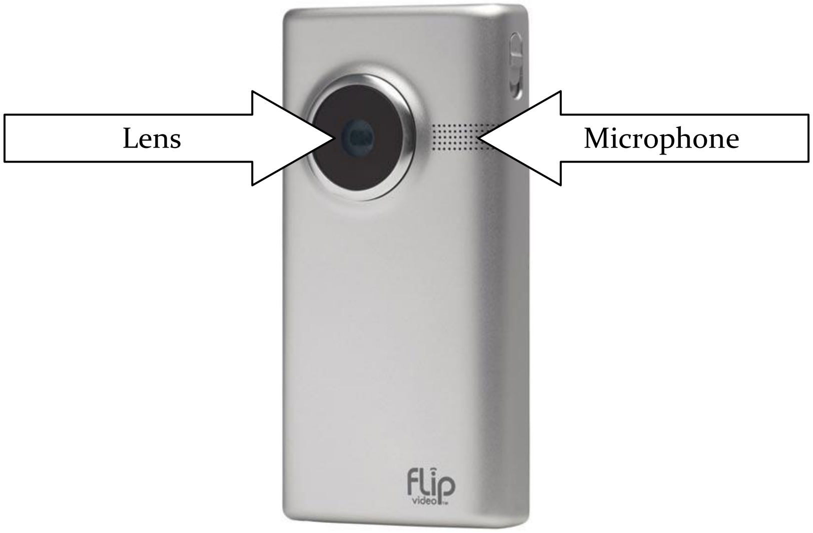


Figure 4.26: The flip cam Mino HD.

In order to leverage the power of this camera, we needed to connect to its control board. To do this we soldered wires to test pads found on the main board. These test points provided logic level access to the record button, the power button, USB power and USB data lines (see Figure 4.27). Wires were soldered to these lines and interfaced directly to the digital ports on the microcontroller. The USB data and power lines were broken out to a USB cable in order to facilitate the download of video data.

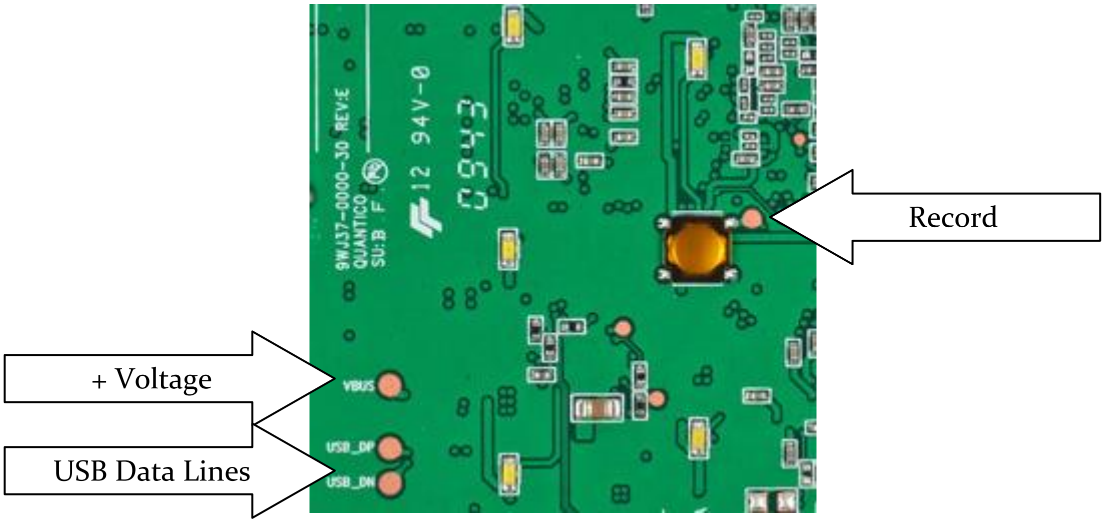


Figure 4.27: Flip cam control board, showing test point connections.

The video quality of this camera is superb; good footage capture leads to a better end video. The video is captured at 30 frames per second (constant frame rate, progressive scan H.264 video compression, AAC audio compression in the MP4 file format. Bit rate is variable at 9.0Mbps based on an auto-adaptive algorithm. Automatic white balance and black level calibration with automatic exposure control with dynamic exposure compensation. The lens is fixed focus (0.8m to infinity) at an aperture of f/2.4 [30]. This provides excellent low light capabilities; this is important due to the variable light nature of the environments the robot encountered (see Figure 4.28). Since we know the minimum focus distance is at 0.8m, we know the minimum distance we need to ensure the user is standing within focus distance from the camera during capture.

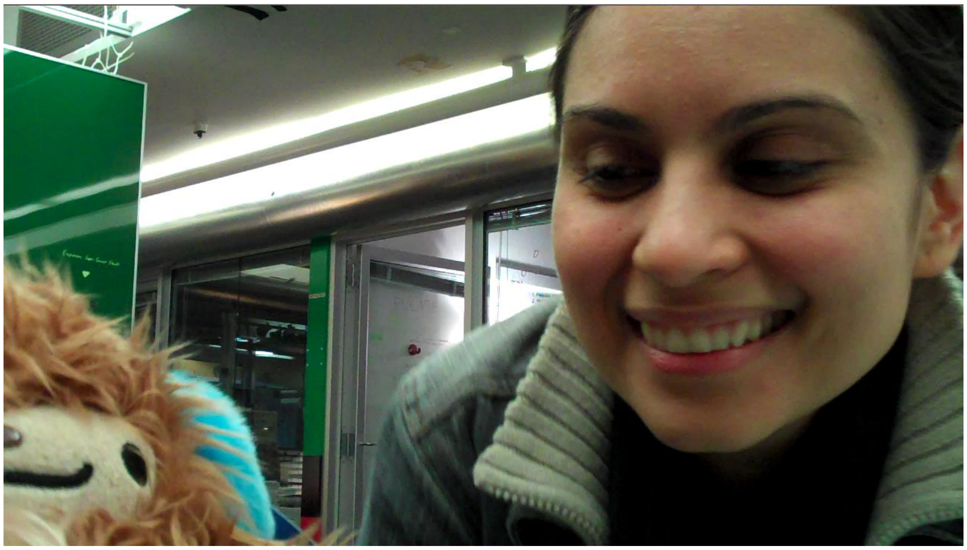


Figure 4.28: Example of a video frame grab from the robot, in a low light condition.

4.2.5 Communications

The robot required two way wireless communications for data logging, location reporting, and control. Wireless LAN was chosen for its ubiquitous nature, reliability and fast communication speeds. To connect to a wireless network from the microcontroller the WiFly GSX (RN-131G) network module from Roving Networks was chosen. This module is a complete low power (100 mWatts) TCP/IP solution ready for surface mounting including a chip antenna (see Figure 4.29). The module consisted of a 2.4GHz radio (802.11b/g), processor, TCP/IP stack, real-time clock, crypto accelerator, power management and analog sensor interfaces. The radio had an effective range of up to 330 feet, 100 meters [31]. The throughput of the device is a nominal 4 Mbps with TCP/IP, this is far above the requirements for this application. The module was connected to the microcontroller through a UART to SPI bridge. This configuration allowed for a faster board level communication speed.

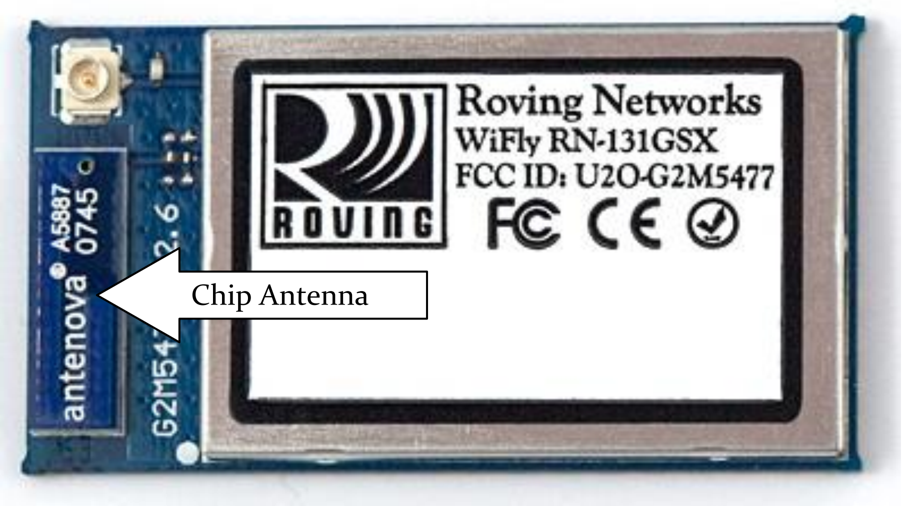


Figure 4.29: Close-up of wifi module RN-131GX.

The robot also utilized ZigBee communications based on the IEEE 802.15.4 protocols. We utilized the NoNo badge platform developed by Nan-Wei Gong as the ZigBee communication module [32]. The robot used this ZigBee protocol to talk to the Ubiquitous Sensor Nodes placed around the Media Lab. The module would send out a unique ID, pinging the nodes as the robot passed by them. Using RSSI information and other methods, the system can determine which node the robot is closest to. Using the information sent by the ZigBee module, we were able to track the robot and capture 3rd person views from the nodes. In this way, we could leverage the robot as an agent for the system.

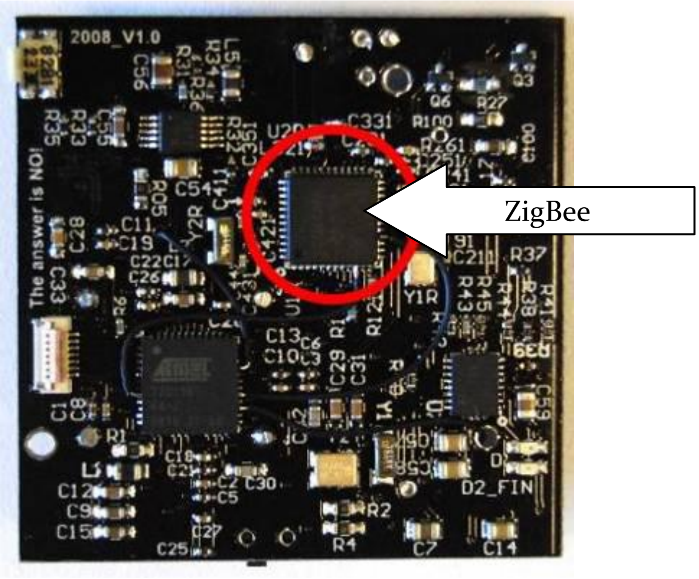


Figure 4.30: ZigBee chip circled in red on the NoNo badge carrier board, used in the robot.

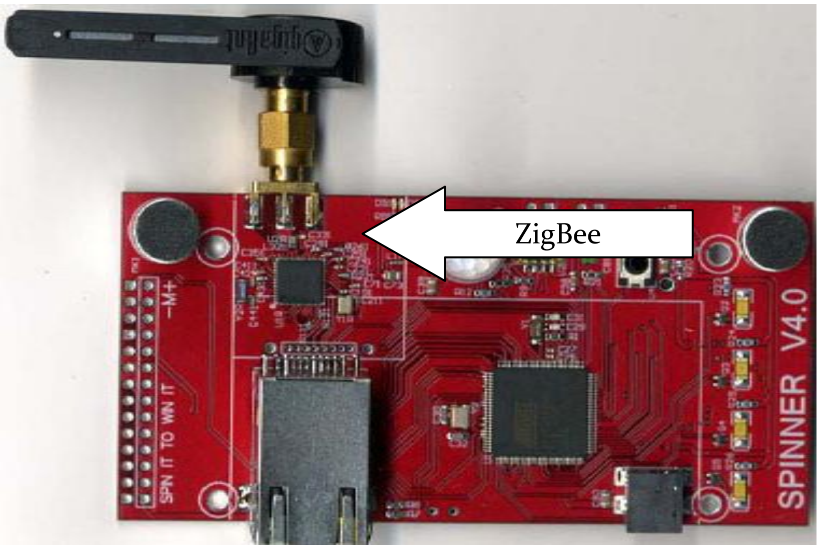


Figure 4.31: ZigBee receiver circuitry on carrier board of a wall mounted portal.

4.2.6 Output

The voice of the robot was recorded as series of MP3 files and edited into blocks. Each one of these blocks were assigned a unique file name and placed on a microSD card. Each block corresponded to a particular part of the robot's script. The file naming convention allowed each file to be individually played by the robot. The board used to play the MP3 files was the MP3 trigger by robertsonics and sparkfun (see Figure 4.32). The board used a Cypress PSoC CY8C29466-24SXI microcontroller and a VLSI VS1053 audio codec; the microcontroller read the MP3 files from the microSD card. The board accepted a 4.5 to 12v input and consumes 45mA idle, 85mA playing. The board was controlled through an SPI interface from the microcontroller and output line level audio [33]. This board was then connected to an audio amplifier, which in turn was connected to the robot's speaker.

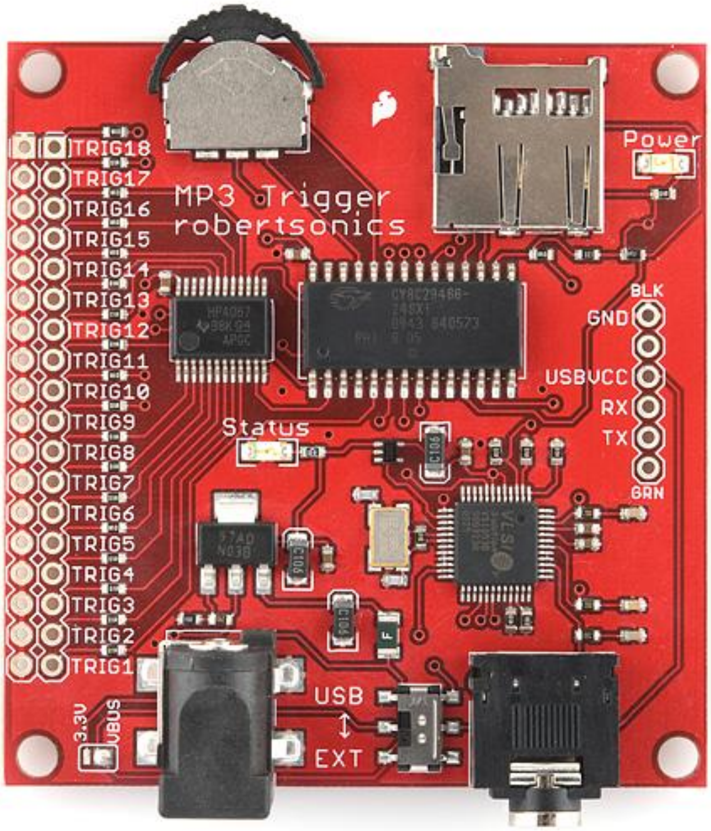


Figure 4.32: MP3 trigger board, used to output the robot's voice.

4.2.7 Locomotion

The robot's motors were controlled by a dual channel H-Bridge chip, the TB6612FNG. This chip supported a motor voltage of 12V while handling 1.2 amp average and 3.2 amp peak draw currents per channel (with thermal shutdown protection). The motor could be put into clockwise, counterclockwise, short break and stop conditions [34]. The chip was controlled by a PWM line from the microcontroller, along with lock lines to control motor direction and stop conditions.

4.2.8 Power

Power was a major concern, since we wanted the robot to be able to rove around for extended periods. We also needed to keep the weight of the robot down to a minimum to enable users to easily carry it. The only type of battery that would allow for a high power density and low weight while also supplying the correct voltage was lithium polymer. The battery used was the Thunder Power 5000mAh LiPoly Triple Cell 3S 11.1V. This battery was connected through a 3A fuse to a master power switch mounted inside the robot's back panel. This switch fed a distribution block, which in turn fed the power to the components.

4.3 Software Design

4.3.1 Firmware

The firmware of the robot consisted of driver libraries for the various peripherals and code to control the interaction and behavior of the robot. The behavior of the robot was controlled by scripting within the code. The hardware interfaces are abstracted from the scripting code. The firmware executed setup and configuration commands, then ran in a master loop (see Figure 4.33). This loop was not terminated until the robot was powered down. The main loop completed a cycle every few milliseconds.

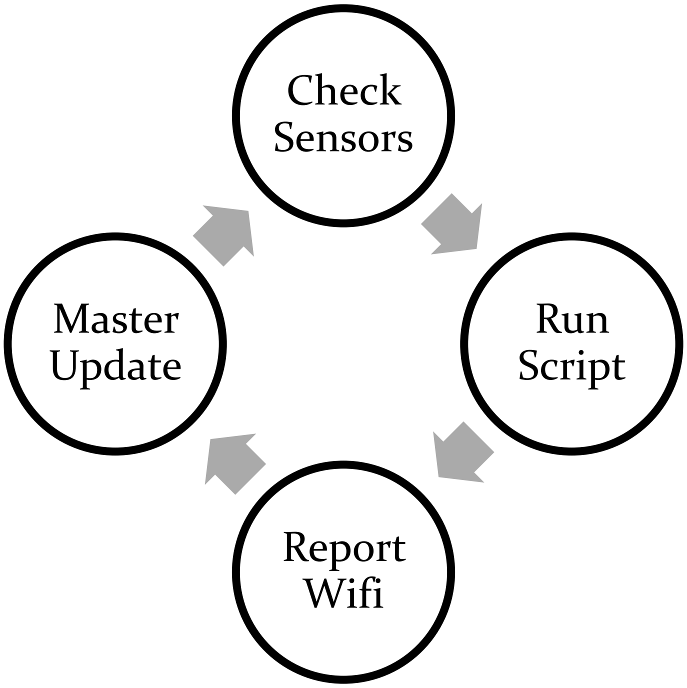


Figure 4.33: Typical master loop in the robot's firmware.

When the robot was powered on, a sequence of configuration and start up commands were executed. First the USB UART was set up at a rate of 115200 baud. This allowed a terminal to connect to the robot to issue debug commands. Next the peripherals were set up. Each required a specific set up sequence to work properly. Setup sequences generally consisted of setting up the proper pins and communication protocols. After the start up sequence, the robot waits to receive input through the USB port. It does this to allow an operator to open up a command line interface for debugging purposes. If no input is detected, the microcontroller proceeds to the master loop.

The master update function updated time sensitive routines within objects in the code. This included updating the command line interface, the wifi transmission code and the mp3 player status. The first task the loop performs is to send out a packet through wifi to the server. The packet contains the start of packet and end of packet delimiters, timing information, the status of the robot sensors, the status of the camera and the current interaction state. This packet is sent to the server and recorded for later analysis and to observe the current state of the robot remotely. The packet structure is shown in Table 4.1.

Table 4.1: Data packet structure.

Datum number	Byte length	Content
0	1	Start of packet delimiter
1	4	Number of master loop cycles since reboot
2	4	Number of milliseconds since reboot
3	2	Current reading from the ultrasonic rangefinder in cm
4	2	Current reading from the infrared distance sensor, in raw ADC value
5	2	Current state of the PIR sensor
6	2	Current raw value of the accelerometer's X axis
7	2	Current raw value of the accelerometer's Y axis
8	2	Current raw value of the accelerometer's Z axis
9	2	Current state of the green button
10	2	Current state of the red button
11	2	Current power state of the camera
12	2	Current recording state of the camera
13	2	Current script question number
14	2	Current program mode
15	1	End of packet delimiter

The scripting routine was run at the current interaction level. Scripting controlled the robot's speech, interaction patterns, camera control and motor control. The first level of the scripting was comprised of human finding routines. Within these routines, the robot attempted to find a human and elicit their help. It did this by controlling the motors and reading the PIR sensor and ultrasonic sensor. The robot roved forward until it approached an object. Once there it performed one of two actions, it acted confused and tried to move away, or intentionally got stuck. If the routine detected body heat within that time, or someone grabbed the robot and moved it, the robot stopped movement and asked the human for help. If the human agreed and pressed the red button, the scripting moved on to the interaction phase. If the human did not press the button, the routine timed out and began the human acquisition process again. If the interaction continued, the robot enters the interaction script routine (see section 3.2.2 "scripting" for a detailed analysis of the script) and started recording. The camera control routine contained the necessary timing and output state to properly simulate the buttons on the camera being pressed. If the human left the robot mid-interaction, the routine timed out, exited the script and began the acquisition process again. During the interaction script, the robot activated the camera to record the interaction.

4.3.2 Server and Video Network

The robot communicated with a central data server to record its interactions and with the Ubiquitous Media Network Portals if within range. The robot data and control server (roboCon or robot control server) took the data packet from the robot over wifi, processed it and recorded a CSV file for later analysis (see Table 4.1 for the data packet contents and structure). This was accomplished through both processing on the microcontroller and processing on the server (see Figure 4.34).

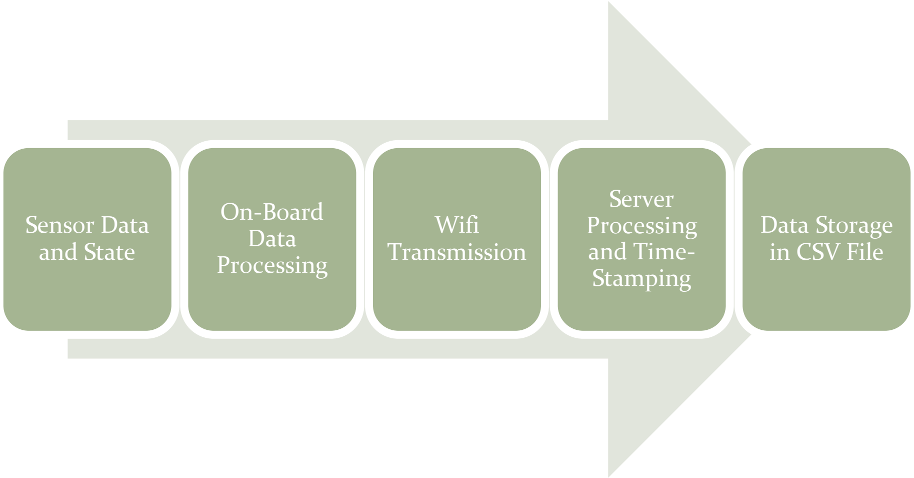


Figure 4.34: Data logging process.

The log file contained data sent from the robot ever 250 milliseconds (4 times per second) (see Figure 4.35). This provided enough resolution to gather data from the sensors without saturating the network or bogging down the microcontroller. The Ubiquitous Media Network tracked the robot and recorded its path in a CSV file on a central server (see Figure 4.36). This CSV file contained a time stamp along with the node ID of the node which the robot was closest to. This node ID could be matched with a node map in order to approximate the robot's location (see Figure 4.37). This data was used for later analysis and evaluation of the system. It also provided a method with which to extract the video form the node closest to the robot. This tracking provided a 3rd person view of interactions which occurred close to a node. Figure 4.38 shows a frame from a video stored on the server, captured from a 3rd person view. The system was able to track the robot and produce media from that view from the data captured.

```
2010,4,20,16,11,32,,_53,13500,145,283,1,364,347,538,1,1,0,0,0,1,_,
2010,4,20,16,11,32,,_54,13750,138,279,1,357,362,508,1,1,0,0,0,1,_,
2010,4,20,16,11,32,,_55,14000,134,277,1,286,328,493,1,1,0,0,0,1,_,
2010,4,20,16,11,33,,_56,14251,128,271,1,364,307,523,1,1,0,0,0,1,_,
2010,4,20,16,11,33,,_57,14500,118,269,1,338,348,490,1,1,0,0,0,1,_,
2010,4,20,16,11,33,,_58,14751,113,262,1,310,343,521,1,1,0,0,0,1,_,
2010,4,20,16,11,34,,_60,15250,101,271,1,329,352,498,1,1,0,0,0,1,_,
2010,4,20,16,11,34,,_61,15500,94,270,1,291,325,503,1,1,0,0,0,1,_,
2010,4,20,16,11,34,,_62,15751,91,273,1,331,357,511,1,1,0,0,0,1,_,
2010,4,20,16,11,34,,_63,16000,81,267,1,351,343,515,1,1,0,0,0,1,_,
2010,4,20,16,11,35,,_64,16250,78,274,1,352,362,525,1,1,0,0,0,1,_,
2010,4,20,16,11,35,,_65,16500,78,268,1,336,367,565,1,1,0,0,0,1,_,
2010,4,20,16,11,35,,_66,16751,96,267,1,308,308,482,1,1,0,0,0,1,_,
```

Figure 4.35: Typical robot status log file snippet from the robot control server. Data includes a time stamp, sensor status and interaction status every 250 milliseconds.

```
11A0D13221971BA8,C3B8,5D73,1111,2010/04/02 17:51:48.160601000
11A0D1347626F4E8,C3B8,5D73,1111,2010/04/02 17:51:58.169249000
11A0D1426EA72F68,C3B8,5D73,1111,2010/04/02 17:52:58.172977000
11A0D1426F5E65C0,C3B8,5D73,1111,2010/04/02 17:52:58.184984000
11A0D1C97ACB1C78,C3B8,FBEE,1111,2010/04/02 18:02:38.197243000
11A0D1CBCEA307C8,C3B8,FBEE,1111,2010/04/02 18:02:48.193837000
11A0D1CE22F5C868,C3B8,FBEE,1111,2010/04/02 18:02:58.198481000
11A0D1D07785A1A8,C3B8,FBEE,1111,2010/04/02 18:03:08.207129000
11A0D1D2CAE386B0,C3B8,FBEE,1111,2010/04/02 18:03:18.195726000
11A0D1D51EF8BD68,C3B8,FBEE,1111,2010/04/02 18:03:28.196337000
11A0D1D7730E3688,C3B8,FBEE,1111,2010/04/02 18:03:38.196965000
11A0D1D9C7239450,C3B8,FBEE,1111,2010/04/02 18:03:48.197586000
11A0D1DC1B3976E8,C3B8,FBEE,1111,2010/04/02 18:03:58.198241000
11A0D1DE6F4F4210,C3B8,FBEE,1111,2010/04/02 18:04:08.198890000
11A0D1E0C3646D10,C3B8,FBEE,1111,2010/04/02 18:04:18.199498000
11A0D1E31779DE60,C3B8,FBEE,1111,2010/04/02 18:04:28.200124000
11A0D1E56B8F20D0,C3B8,FBEE,1111,2010/04/02 18:04:38.200738000
11A0D1E7BFE23F30,C3B8,7B73,1111,2010/04/02 18:04:48.205406000
11A0D1EA13F75A90,C3B8,7B73,1111,2010/04/02 18:04:58.206010000
11A0D1F10F805118,C3B8,7B73,1111,2010/04/02 18:05:28.195871000
```

Figure 4.36: Typical robot tracking log file snippet from the Ubiquitous Media Nodes. Data includes a time stamp, robot ID and closest node ID.

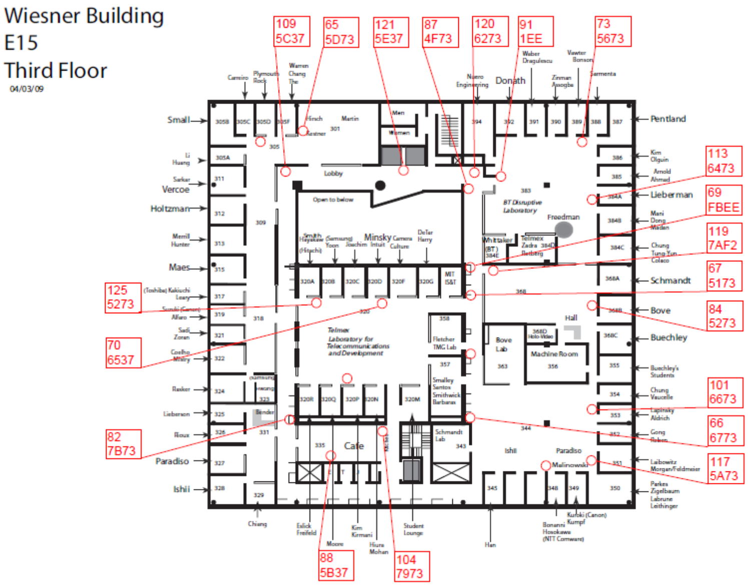


Figure 4.37: Example map of node locations and IDs.

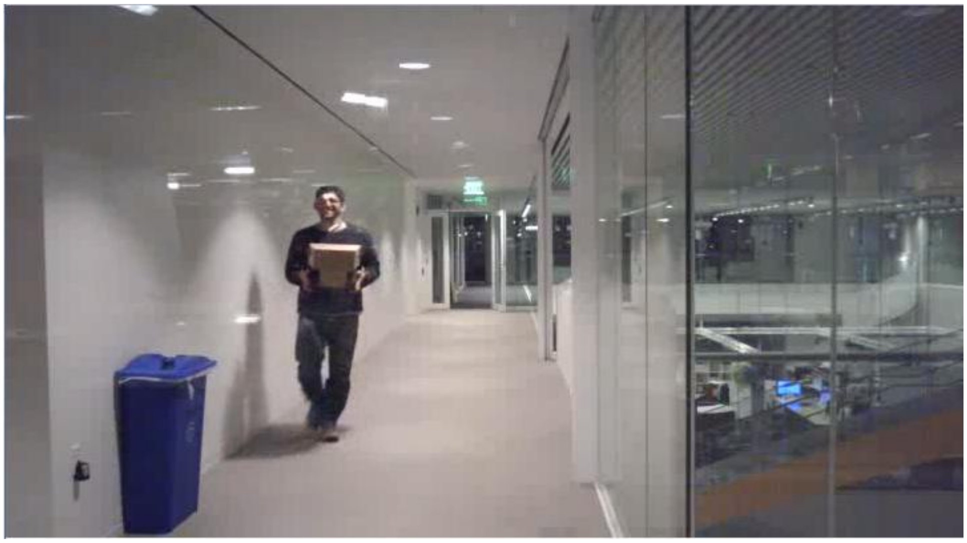
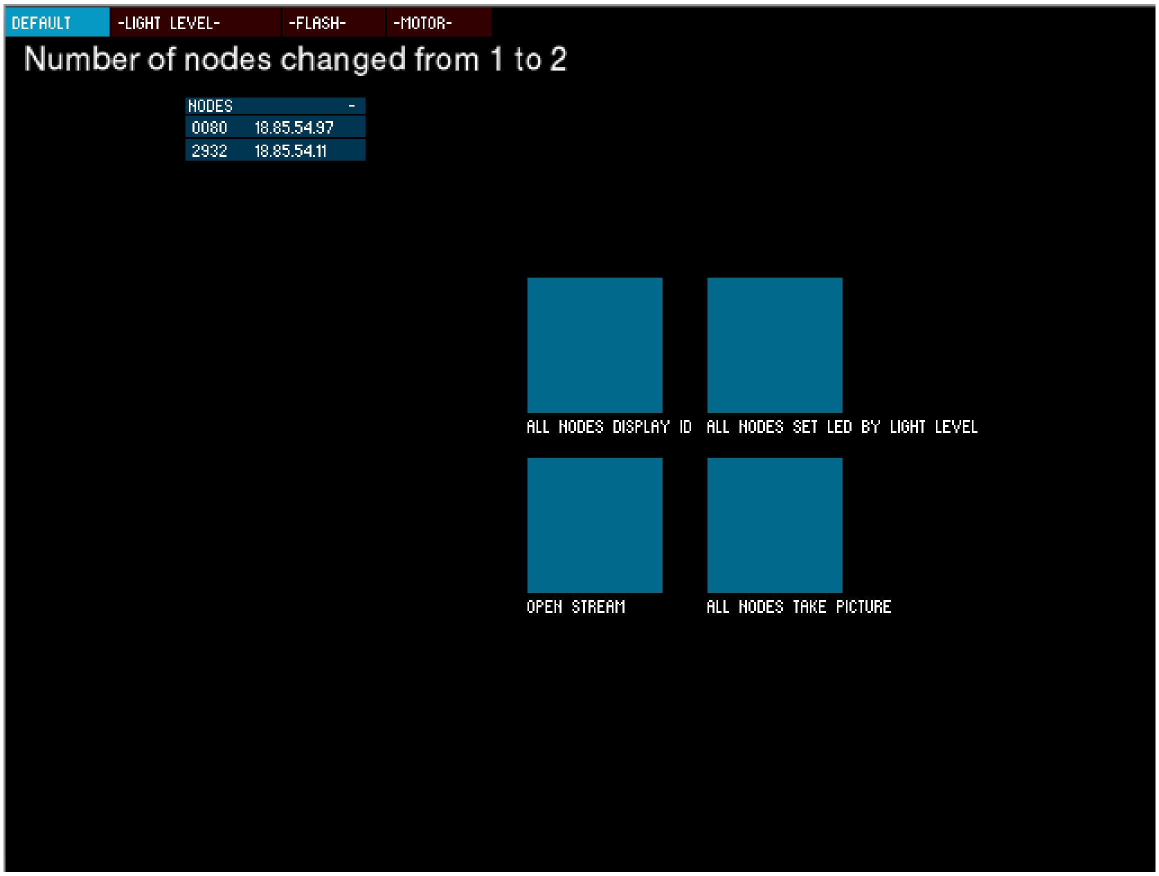


Figure 4.38: Frame grab of a 3rd person view of an interaction. Video captured from a wall-mounted media capture portal.

The ubiquitous media capture portals had to be set up in order to capture video. This was done through a program that sent commands to the nodes telling them how to configure. Configuration included aspects such as camera angle and lighting conditions. Since there were a large number of nodes to configure, setting each one was not practical. So a GUI was developed to set up the system for capture.

Figure 4.39 shows the fist GUI page of SPINNERcon. This page shows currently active nodes, which can be selected and global options. “ALL NODES DISPLAY ID” causes all nodes currently reporting to display their ID on their screens. “ALL NODES SET LED BY LIGHT LEVEL” causes all nodes to set their onboard lights to a level proportional to how dark their current location is. “OPEN STREAM” opens the video stream of the currently selected node, to see its real time status. “ALL NODES TAKE PICTURE” causes all nodes to take pictures and store those pictures on the server.

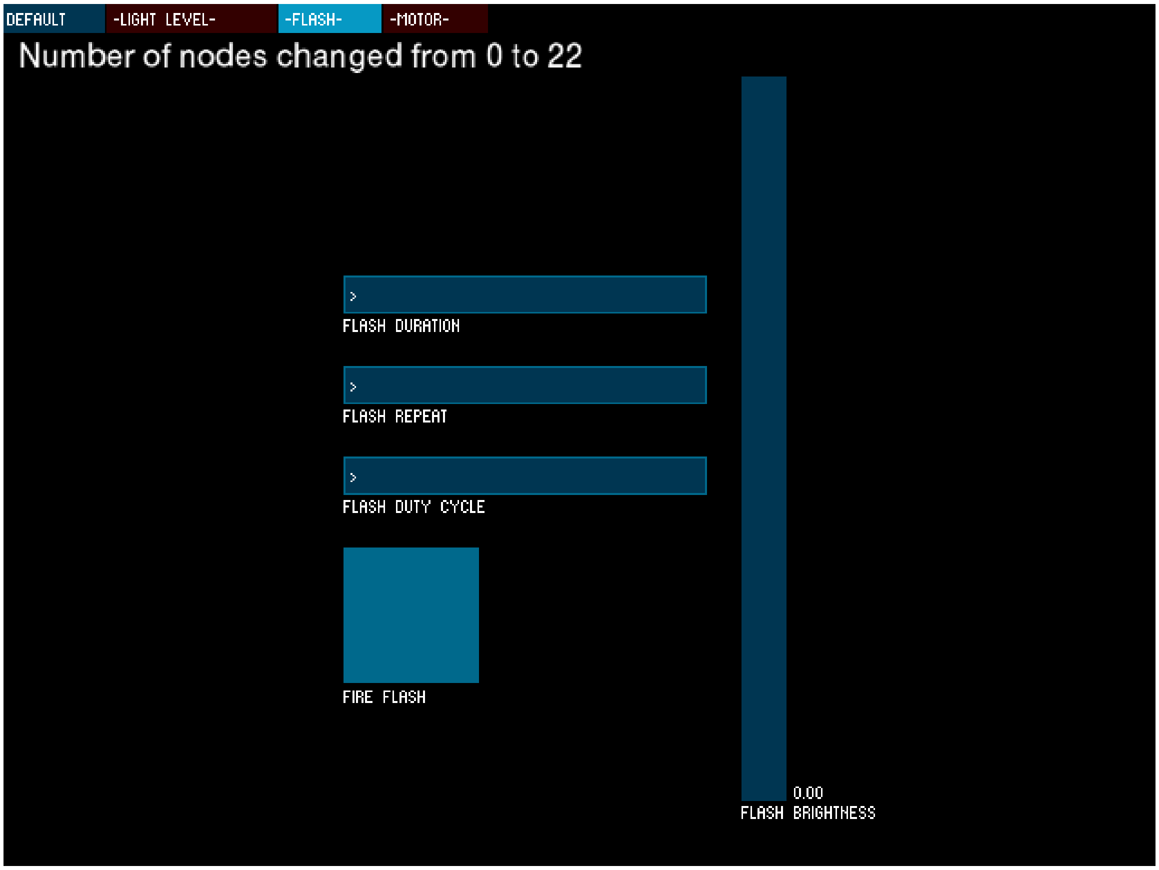


Number of nodes changed from 1 to 2

NODES	-
0080	18.85.54.97
2932	18.85.54.11

Figure 4.39: SPINNERcon GUI showing currently active nodes and options.

Figure 4.40 shows a setup page that configures the node's onboard lights to flash. This was used as a warning to indicate that the system was about to record video.



Number of nodes changed from 0 to 22

>
FLASH DURATION

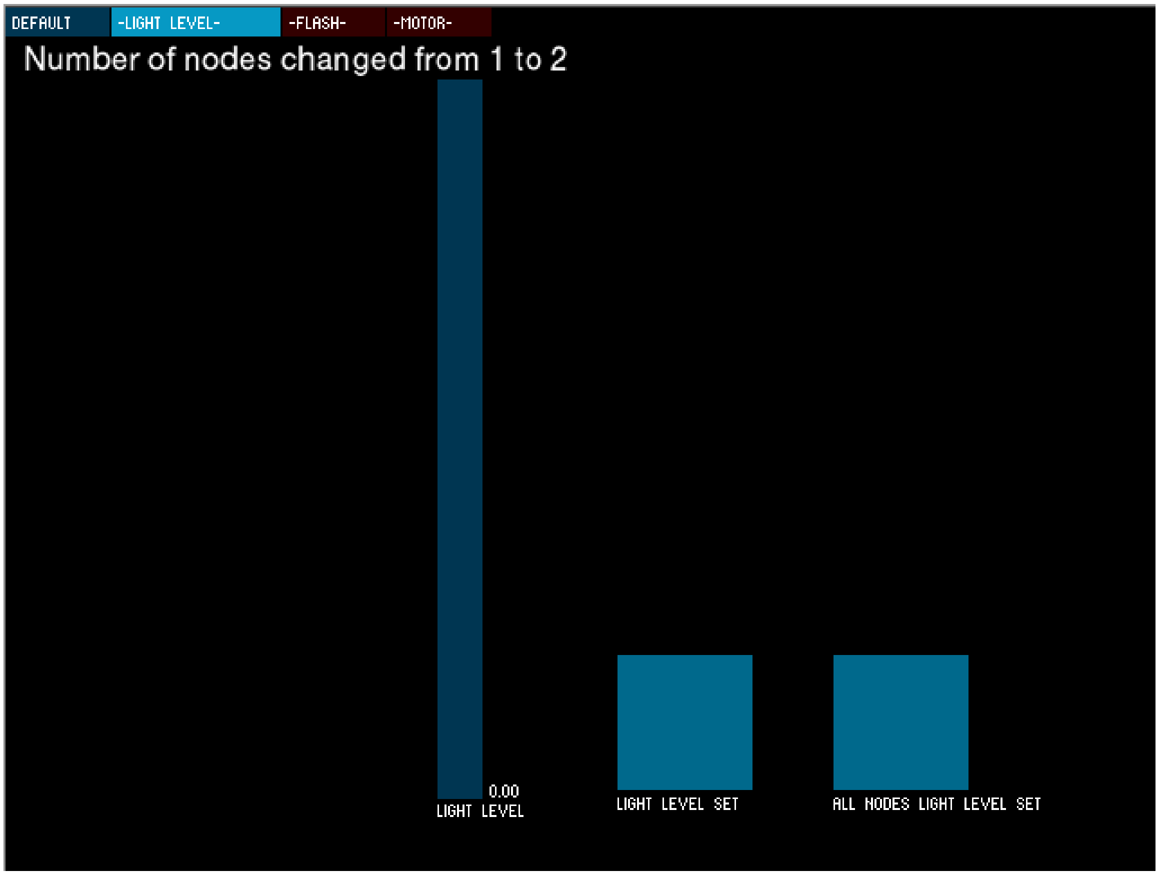
>
FLASH REPEAT

>
FLASH DUTY CYCLE

0.00
FLASH BRIGHTNESS

Figure 4.40: SPINNERcon flash setup page.

Figure 4.41 shows a setup page that configured the onboard light level. This allowed for fine grain control of the node’s light level if the automatic configuration was off.

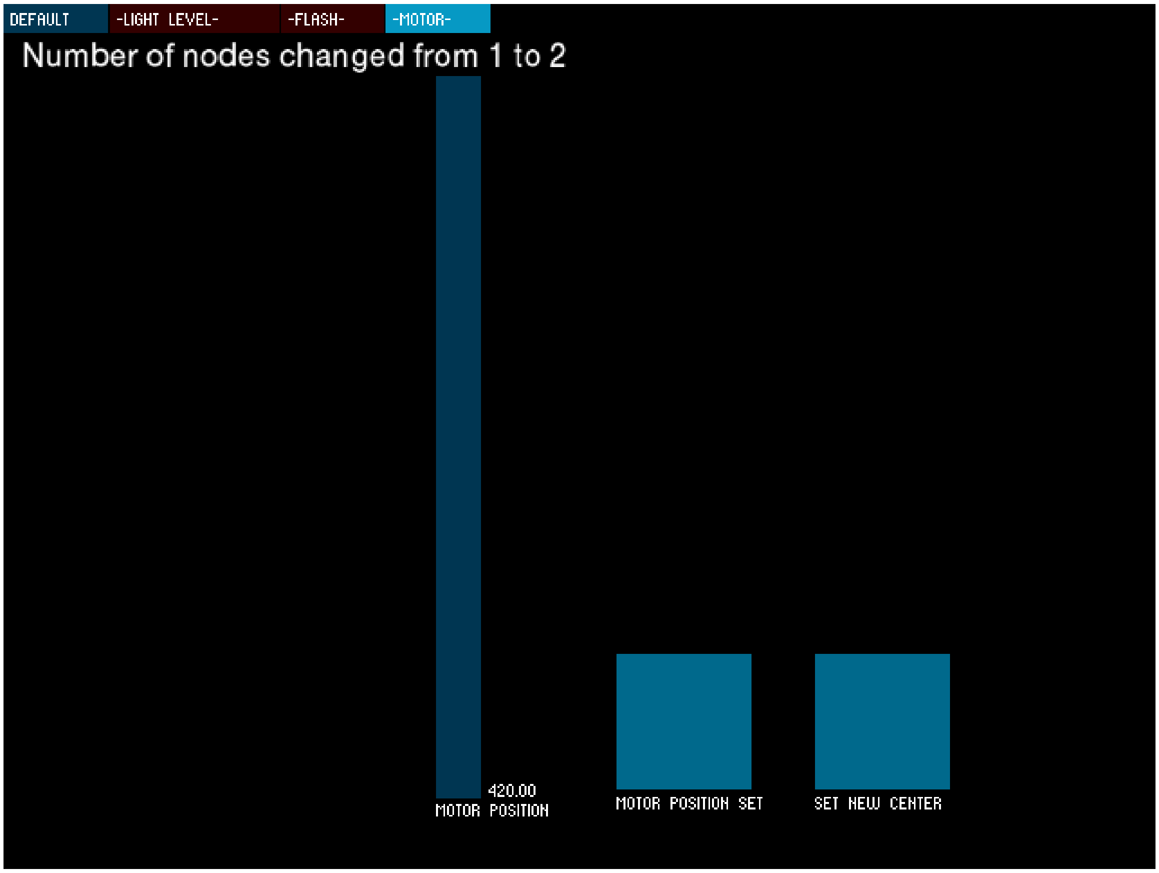


Number of nodes changed from 1 to 2

0.00
LIGHT LEVEL

Figure 4.41: SPINNERcon light level setup page.

Figure 4.42 shows a setup page that configures the camera angle of the node. Each node's camera had to be configured manually to obtain the best capture angle. This was due to the uneven node mounting height and unknown node position on the wall.



Number of nodes changed from 1 to 2

Figure 4.42: SPINNERcon camera angle setup page.

Chapter 5

Results and Evaluation

5.1 Evaluation

We studied the best methods and interaction scenarios with which to leverage human interactions into the creation of stories. As it is difficult to qualitatively analyze the quality of a story, other metrics for evaluation were chosen. Times of interaction, script types, location of the robot vs. effectiveness and user surveys were evaluated. A metric for "complete interaction" vs. "incomplete interaction" was formed with which to evaluate the data with. As the robot was able to produce a subjectively entertaining and informative story in video form, the system was deemed a subjective success. While not all of the original goals of the system were explored, the system was able to complete the major tasks assigned to it. Future work is discussed in the evaluation where appropriate.

5.2 Story Capture

5.2.1 Introduction

Data for evaluation was captured in several formats. These formats included data packets from the robot during real time interaction (see section 4.3.2), length and content of videos captured, data from the Ubiquitous Sensor Portals and user surveys. We were able to capture 50 valid interaction events from which we could extract viable data. As the stories gathered by the robot were in video form, it is difficult to include this data in this thesis. However, some examples will be discussed and full interaction is shown in Appendix A: Annotated Video Stills. We analyzed the time of interaction along with other factors to evaluate the quality of interaction obtained.

5.2.2 Overall Interaction Time

Figure 5.1 shows the average percent of interactions (of the two script types) captured verses the time spent interacting with the robot. The minimal amount of time to complete the full useable interaction sequence was a little over 3 minutes. A full useable interaction sequence was defined as interaction with the robot to the point of being a useable narrative (over 90% question completion rate). In Figure 5.1 we can see a clear separation between full interaction and partial interaction at the 2.5 to 3 minute mark. We used this mark to identify the videos which were full useable interaction vs. partial interaction. There exists a large spike of interactions which lasted for less than 1 minute. The majority of these interactions were found to be, through analysis of the video and question progression, either users pushing the button and walking away or not choosing to continue interacting with the robot after quickly investigating it. It was found the 70% of the interactions fell into the "partial interaction" category while 30% fell into the "full useable interaction category". Note that it was to be expected that the number of complete interactions would be low. This was due to the fact the complete interactions took a disproportionally large amount of time vs. incomplete interactions. Since the time the robot was roaming was finite, more incomplete interactions could be recorded vs. the number of complete ones. Furthermore, complete interactions implied a full commitment to the robot, which required more time of the participant.

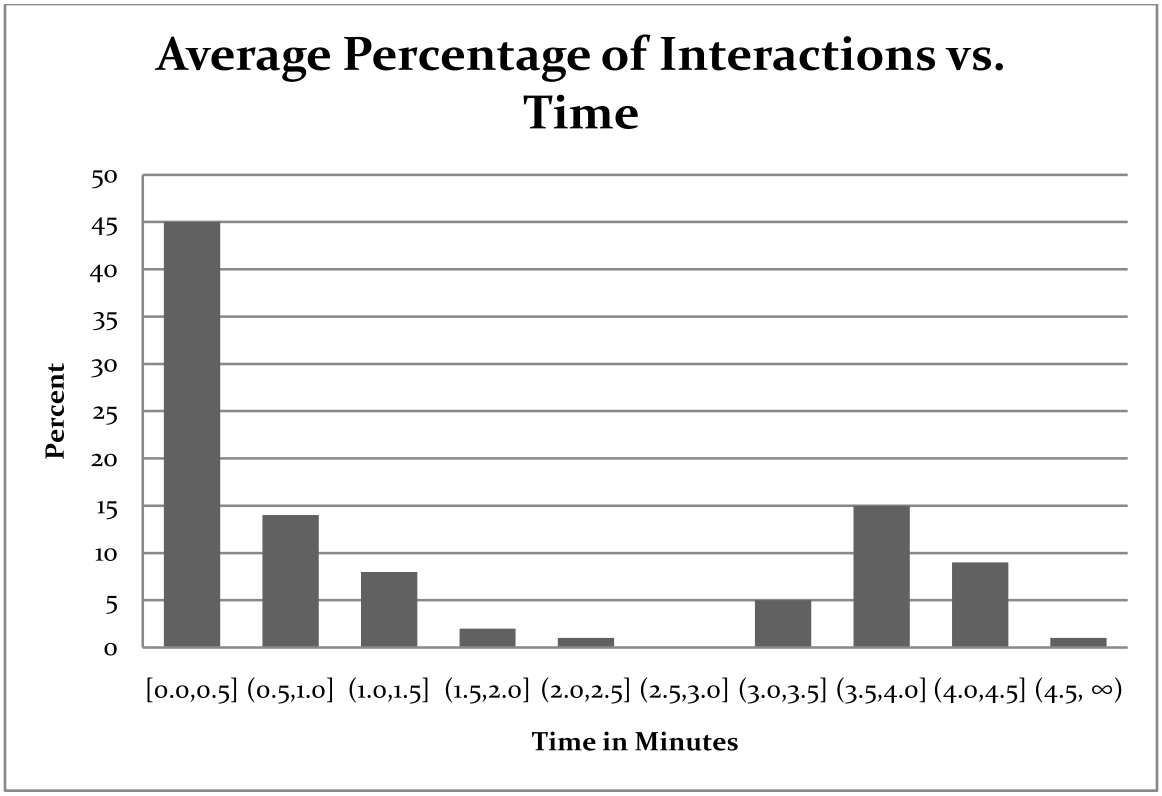


Figure 5.1: Average percentage of interaction vs. time.

5.2.3 Quality of Interactions

Figure 5.2 shows percentage of interactions vs. time for the different script types produced. Script A consisted of simple commands while Script B tried to simulate a two-way conversation with the user (see section 3.2.2 for more discussion). Note that in the entry stage of the interaction the two scripts are similar. This indicated that regardless of what the robot says, most users simply press a button and walk away. However, after about 45 seconds, the scripts differ and fewer users abandon the robot in this period with Script B vs. Script A. While Script A loses users quickly (most after 1.5 minutes) Script B holds user's interest longer, losing none after 1.5 minutes of interaction until they complete the interaction. After the 3 minute "complete interaction" threshold, Script B has 36% more users still interaction with the robot vs. Script A. Only 12% of the total users attained "complete interaction" after starting with the robot, with Script B keeping 48%. Script B was extremely effective for producing "complete interactions". This conclusion is supported by Brickmore et. al who determined that "social dialogue was demonstrated to have an effect on trust, for users with a disposition to be extroverts." [21]

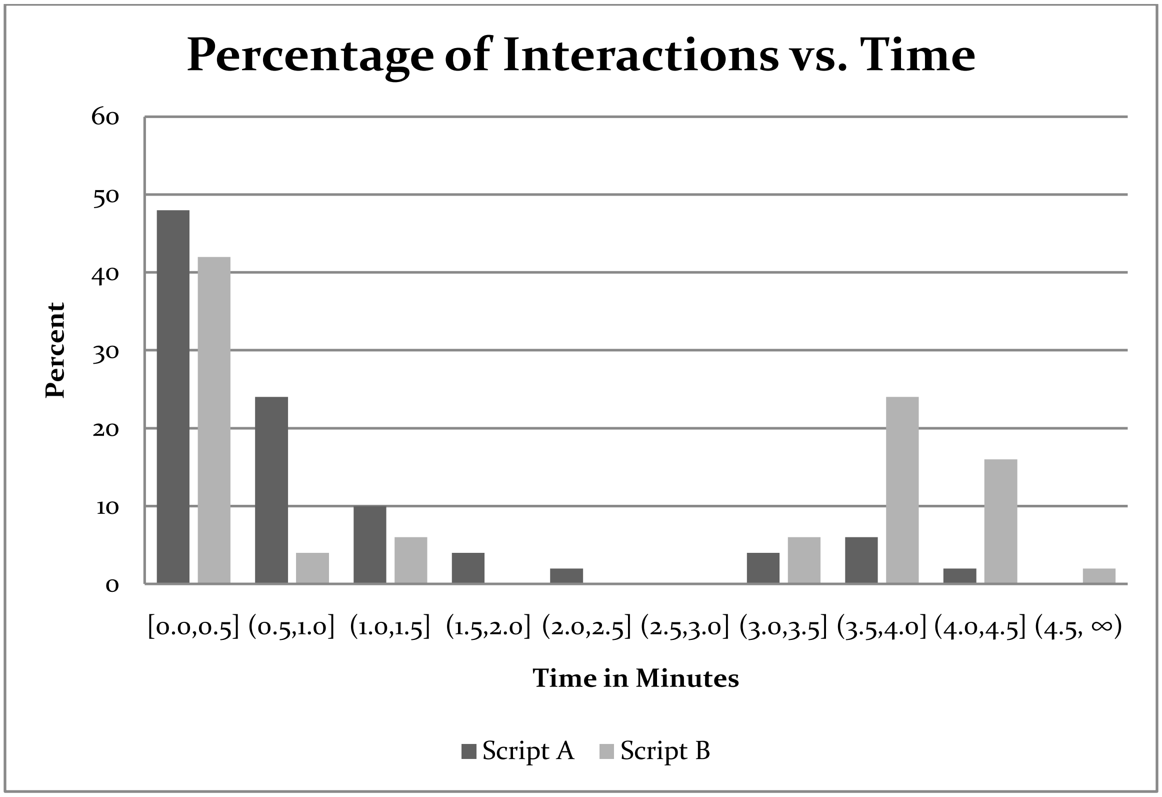


Figure 5.2: Percentage of interaction vs. time for different script types.

Figure 5.3 shows how many of the total interactions took place in which areas. There were three types of areas; a public area was an area where everyone had access but was not a through way for walking (i.e. an atrium but not a elevator), a private area was an area where only people who needed to be there traveled (i.e. an office area), and a transitional area was a public area used mostly for getting from one place to another (i.e an elevator or hallway). This graph shows several important points. One is that while public places had the most interaction, compared to private places the ratio of interactions vs. complete interactions was lower (the percentage of all interactions in that area vs. the percentage of complete interaction). In public places there was a 33.3% rate of interaction "completeness", in private places it was 66.6% and in transitional areas it was 16.5%. Therefore, while there were fewer interactions in private places, the number of "complete interactions" vs. total interactions was higher. This shows that private places produce more "complete interactions" vs. total interactions, but will not capture as many in the same amount of time. The graph also shows while transitional areas had a lot of initial interaction, these areas did not produce many "complete" interactions. This was due to the fact that people in those areas are moving from one place to another and may not have time to interact with the robot (as determined by a subsequent survey).

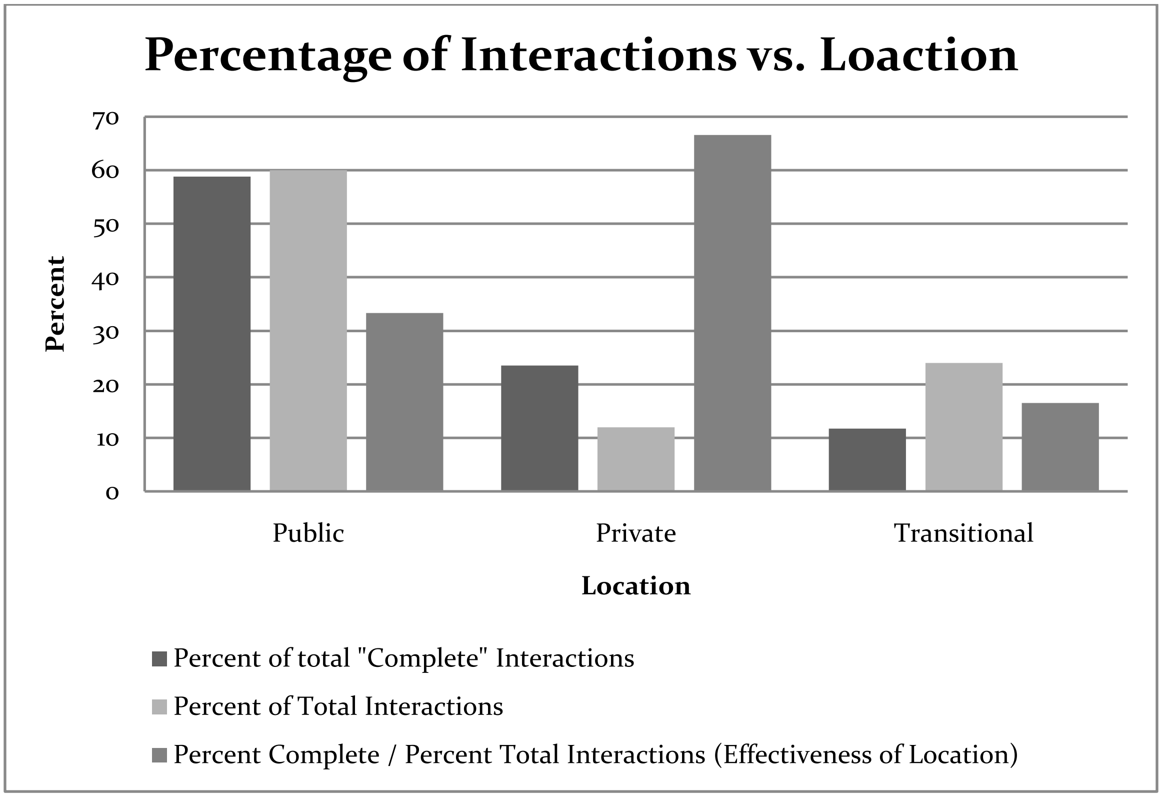


Figure 5.3: Percentage of interaction vs. location of interaction.

Figure 5.4 shows the percentage of total daily interactions vs. the time of the day. Notice the spikes around lunch time and around 4pm. By observation of the lab during these times, it was found that most social activity occurred during these periods. People would often gather to eat lunch together and take a break around 4pm. Using this information for this particular area, the robot was deployed between the hours 11 and 18. This yielded the most interactions and was the most efficient time to gather stories.

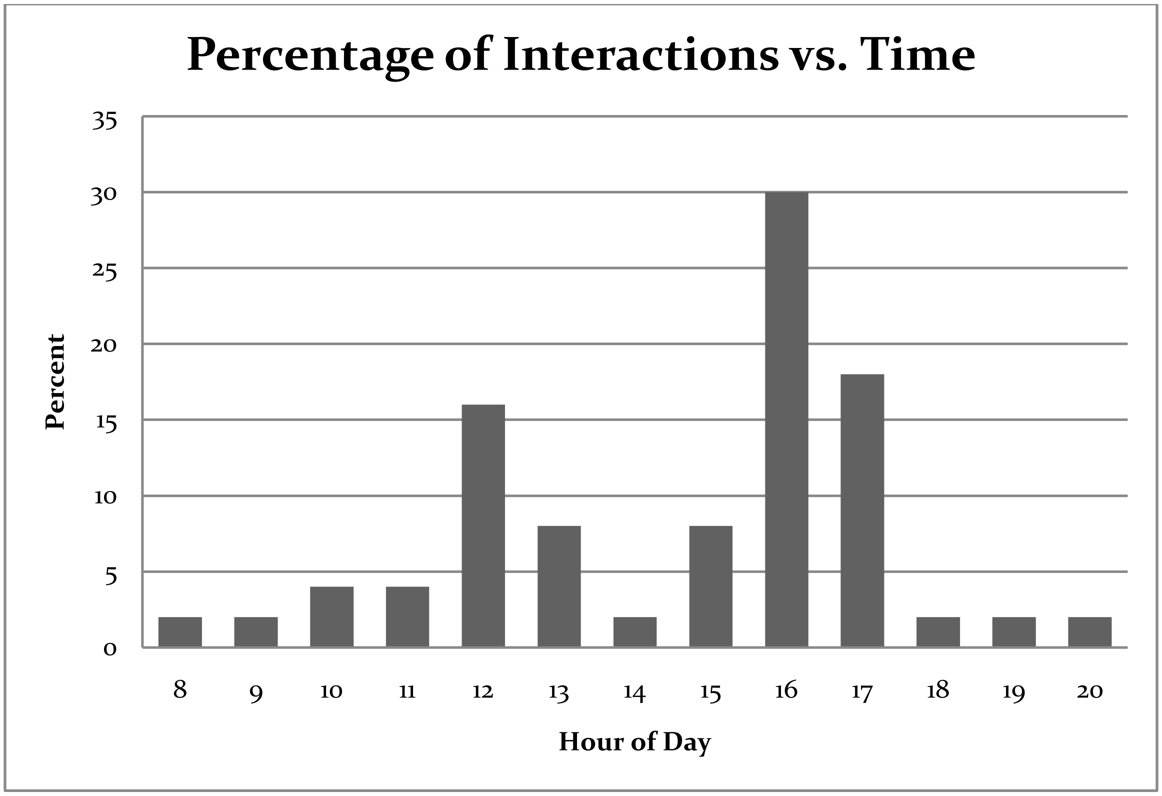


Figure 5.4: Percentage of daily interactions vs. time of day.

5.2.4 Ubiquitous Sensor Portal Capture

As discussed in Section 4.3.2, when in range, the robot leveraged the Ubiquitous Sensor Portals which were deployed around the lab. The main problem encountered when using this portal was that the interactions with the robot often did not occur in their view. This also illustrates one of the shortcomings of a wall mounted system such as this. However, we were able to capture some video from a 3rd person view from the portals along with 1st person views of the robot. Figure 5.5 and Figure 5.6 show a 3rd person view and a 1st person view of the same interaction event. These two images occur at the same time in the same place, as indicated by the ZigBee communication packet. The videos obtained consisted mostly of users carrying the robot to its destination rather than capturing a full interaction event.

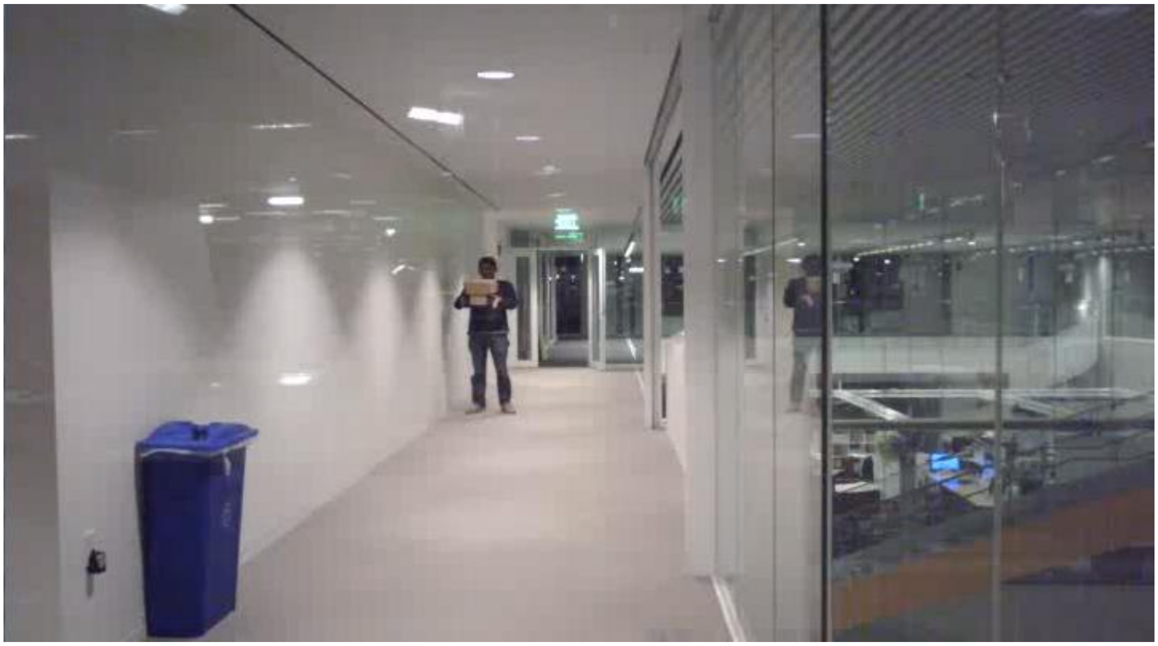


Figure 5.5: A 3rd person view of an interaction, seen from a wall mounted sensor portal.

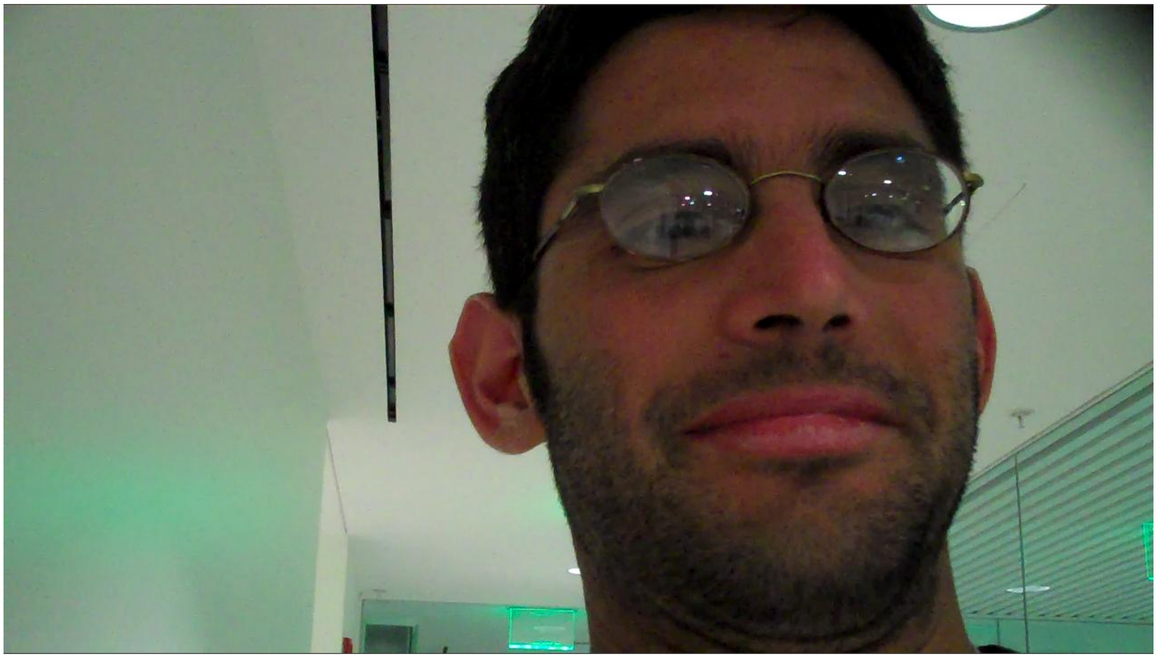


Figure 5.6: A 1st person view from the robot.

5.3 User Survey

After running all tests with the robot, a user survey was sent out to a public mailing list in the Media Lab. This survey asked various questions regarding the robot. Two surveys were offered, one for those who interacted with the robot and on for those who saw the robot but chose not to interact with it. The questions were answered on a scale from strongly disagree to strongly agree.

5.3.1 Interaction Survey Results

28% of the survey participants (7 people) did not complete the full interaction with the robot and 72% of the participants (18 people) had completed the full interaction. There were 10 people who responded to the "no interaction" survey.

- **Question 1** , Figure 5.7:
  "I felt compelled to help the robot"

In this question we wanted to determine if the robot successfully created a link with the user. The majority of users who did not complete the full interaction did not feel they needed to help the robot, while the majority of users who completed the full interaction indicated that they did feel like they needed to help. There were a small percentage of users who did not complete the full interaction but felt compelled to help the robot; however they had other reasons for not completing interaction, as indicated by the survey questions. Over 70% of the responders said they felt compelled to help the robot. This indicated that the robot was successful in displaying helpless behavior that prompted empathy most users.

- **Question 2,** Figure 5.8:
  "I thought I would feel guilty if I did not help the robot"

In this question we wanted to see of the user was motivated by guilt to help the robot. Over 60% of the users disagreed while none strongly agreed. The users which did not complete the full interaction tended not to feel guilt for not helping the robot. This data may indicate that the robot needed "cuter" features to elicit guilt. Future work would include testing different robot configurations and their effect on guilt.

- **Question 3,** Figure 5.9:
  "I interacted with the robot because it asked for help"

In this question we wanted to determine if asking for help was an effective interaction tactic. The data shows that 80% of the users helped the robot because it asked for assistance. This would seem to indicate a strong linguistic tie to the interaction.

- **Question 4,** Figure 5.10:
  "I interacted with the robot because it looked interesting"

In this question we wanted to determine if the physical appearance of the robot influenced interaction. 80% of the users strongly agreed that the robot's "interestingness" was what led them to interact with it. This supports the hypothesis that the appearance and behavior of the robot is an important factor in user acquisition. As discussed in "How to approach humans?: strategies for social robots to initiate interaction" a robot that appears to engage a subject is more successful in capturing their attention [19].

- **Question 5,** Figure 5.11:
  "I interacted with the robot because I wanted to participate in the movie"

In this question we wanted to determine if users interacted with the robot to be in a publicized movie. Over 95% of the users did not care or did not interact with the robot because of this. This shows that the robot itself was enough to elicit interaction without the promise of an appearance in a movie.

- **Question 6,** Figure 5.12:
  "I felt that the robot was using me"

In this question we wanted to determine if the robot successfully attained its goals without leaving the user feeling "used". Over 90% of the users did not care or did not feel that the robot was using them. This indicates a success in the interaction design of the robot, since our goal was to not appear like we were exploiting the user. Those who did feel like the robot was using them did not complete the full interaction. Brickmore et. al. found that "small talk" increased trust of an agent [21]. The fact that the users did not feel that the robot was using them was an indicator of the successful nature of the social language employed.

- **Question 7,** Figure 5.13:
  "I felt I knew the purpose of the robot"

In this question we wanted to determine if the robot communicated its intent. The results here were fairly even, although a large percentage of those who did not know the purpose of the robot were those who did not complete the full interaction. The results are fairly surprising, since the robot stated its full intention verbally after the initial interaction. This may indicate that those who did not fully interact with the robot did not listen to it carefully or were unclear of its intent. It was important the users knew the intent of the robot as Cassell et al found that "the use of social language was demonstrated to have a significant effect on users' perceptions of the agent's knowledgableness and ability to engage users, and on their trust, credibility, and how well they felt the system knew them, for users manifesting particular personality traits." [35]

- **Question 8,** Figure 5.14:
  "I felt that interacting with the robot was enjoyable"

In this question we wanted to determine if the robot successfully created a link with the user. All responses that indicated that the users who did not enjoy interacting with the robot were from those who did not complete the full interaction. Those who completed the full interaction found the interaction with the robot enjoyable enough to hold their attention. This may indicate that the interaction should be tweaked to make the interaction more enjoyable, since even though mot who completed the full interaction agreed that it was enjoyable the majority did not strongly agree.

- **Question 9,** Figure 5.15:
  "I felt the interaction with the robot took too long"

In this question we wanted to determine if the robot successfully created a link with the user. Over 80% of users agreed that the interaction took too long. The average full interaction time was about 4-5 minutes. This indicates that this type of system can only leverage a human for less than 4 minutes before they feel that the interaction in a burden. Future research can explore what types of interaction may be too long and how to effectively dynamically change the interaction scenario.

5.3.2 Non-Interaction Survey Results

- **Question 1,** Figure 5.16:
  Why did you not interact with the robot?
  "I felt that the robot was annoying"

There was a strong split between users who did not interact with the robot, where 40% agreed and 40% disagreed. This shows that the robot's level of "annoying" qualities had more to do with user perception than the robot as there was no agreement. It is hard to determine what will annoy some and appeal to others, as it is a matter of personal preference.

- **Question 2,** Figure 5.17:
  Why did you not interact with the robot?
  "I felt that the robot had an ulterior motive"

Most who did not interact with the robot thought it had an "ulterior motive". This indicates that the robot seems suspicious and untrustworthy. This does not agree with Figure 5.12, where most users did not feel that the robot was exploiting them. This could indicate a suspicious nature in those who chose not to interact with the robot. Further study needs to be done to determine which features of the robot lead to this suspicion.

- **Question 3,** Figure 5.18:
  Why did you not interact with the robot?
  "I did not have time to interact with the robot"

70% of users strongly agreed that they did not have time to interact with the robot. This is the strongest reason for non interaction found. This may be due to the user seeing the robot in passing or having other business to attend to, not bothering to interact. There is not much in the robot's design that can change this, as business of the user is completely external.

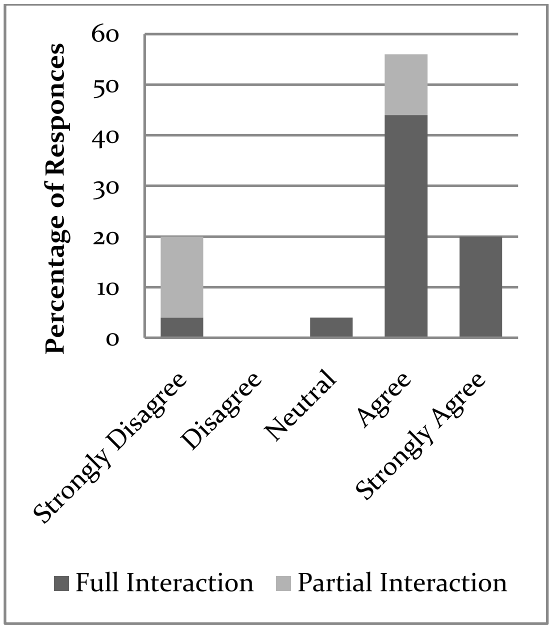


Figure 5.7:
"I felt compelled to help the robot"

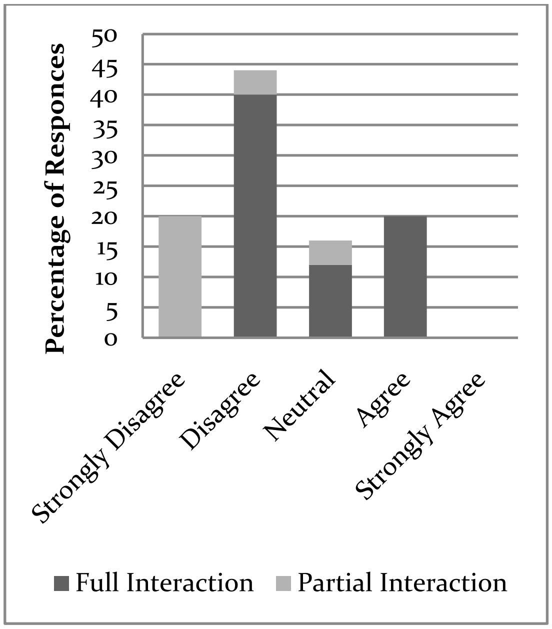


Figure 5.8:
"I thought I would feel guilty if I did not help the robot"

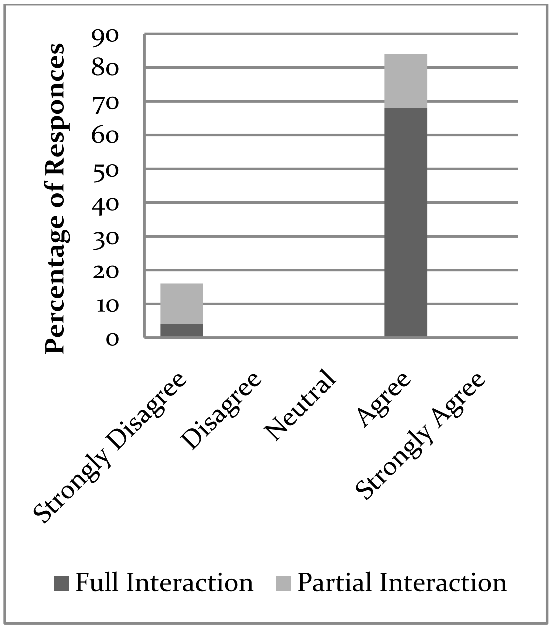


Figure 5.9:
"I interacted with the robot because it asked for help"

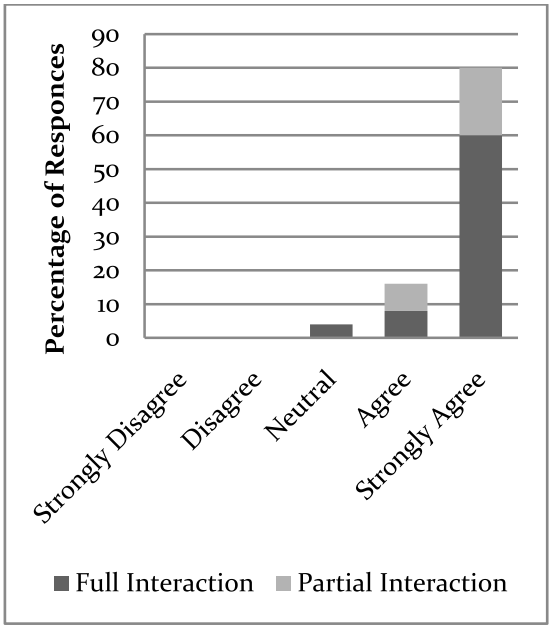


Figure 5.10:
"I interacted with the robot because it looked interesting"

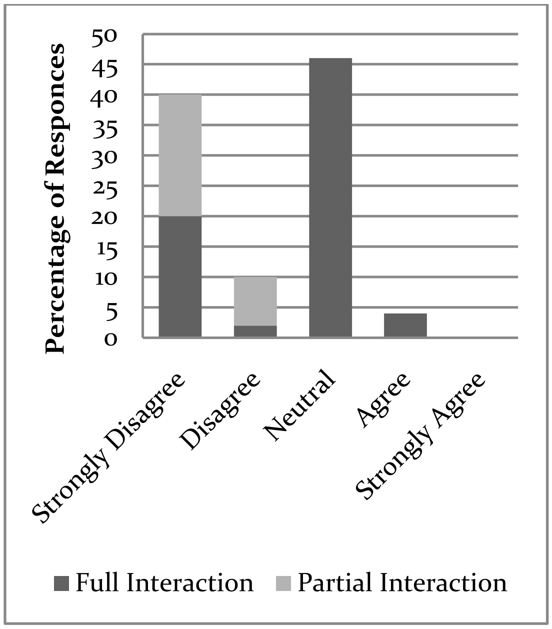


Figure 5.11:
"I interacted with the robot because I wanted to participate in the movie"

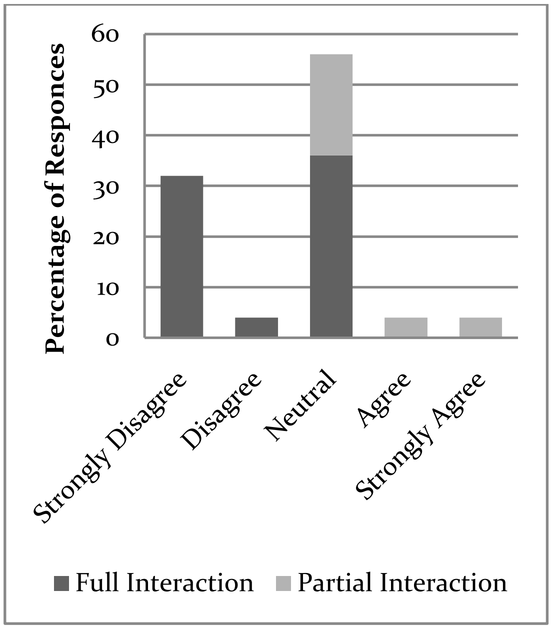


Figure 5.12:
"I felt that the robot was using me"

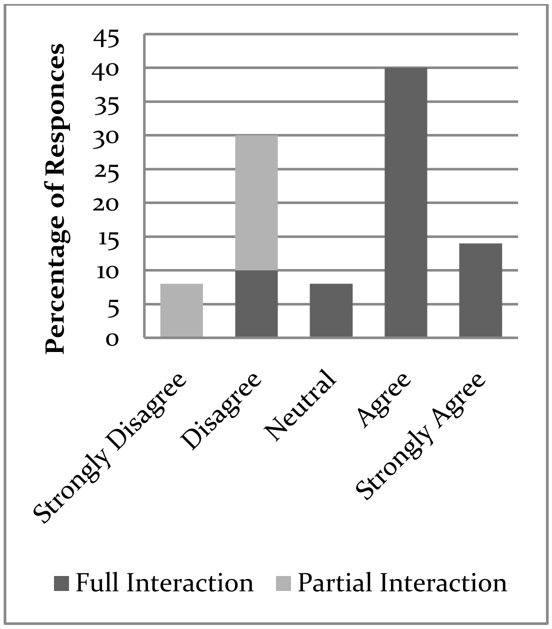


Figure 5.13:
"I felt I knew the purpose of the robot"

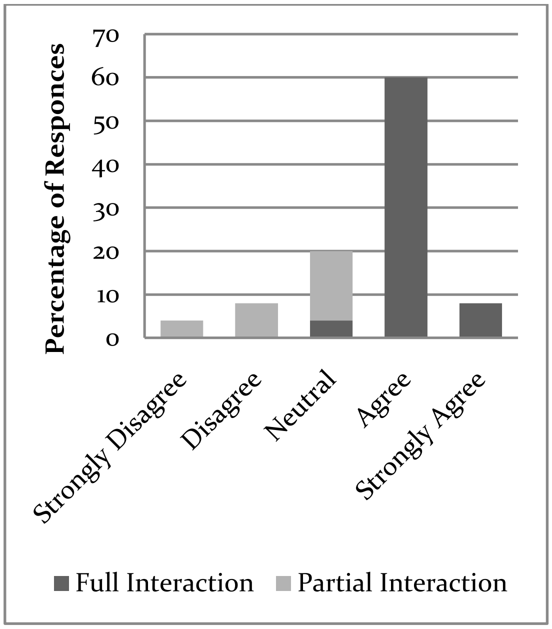
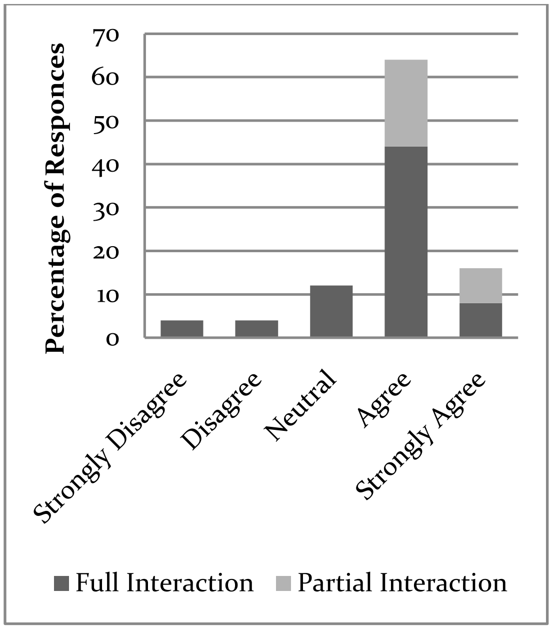


Figure 5.14:
"I felt that interacting with the robot was enjoyable"



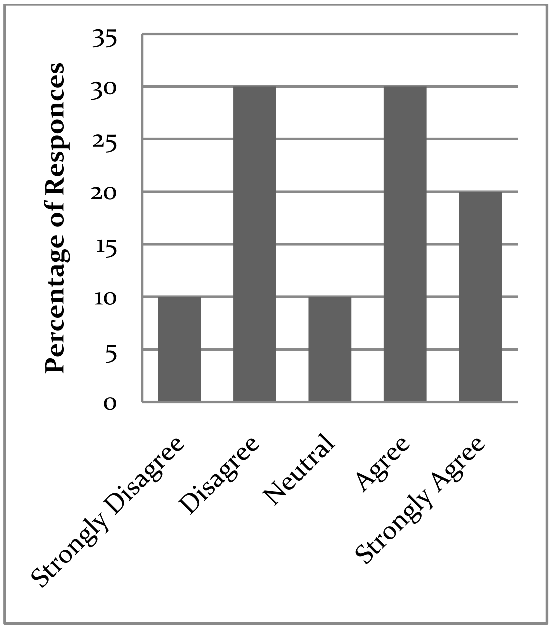
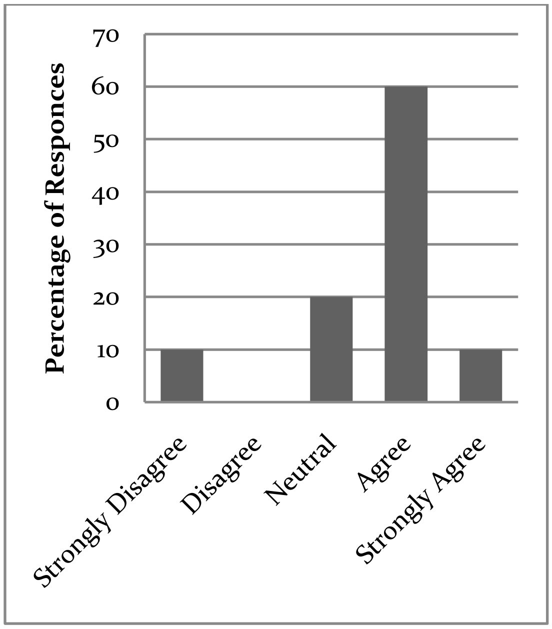


Figure 5.15:
"I felt the interaction with the robot took too long"

Figure 5.16:
Why did you not interact with the robot? "I felt that the robot was annoying"



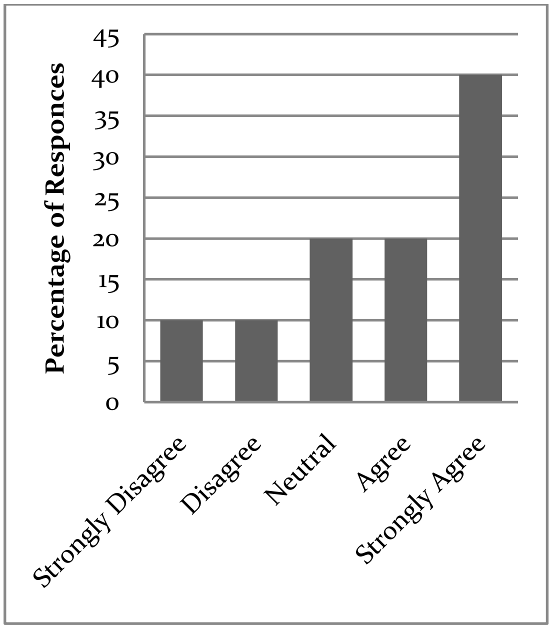


Figure 5.17:
Why did you not interact with the robot? "I felt that the robot had an ulterior motive"

Figure 5.18:
Why did you not interact with the robot? "I did not have time to interact with the robot"

5.3.3 Other Factors

While analyzing the survey data, an unexpected and interesting trend emerged was those who indicated partial interaction with the robot were the same as those who did not take the time or effort to complete the survey completely (see Figure 5.19). Figure 5.19 shows the percentage of respondents who listed personal information vs. robot interaction type. It can be clearly seen that the type of person who would interact with the robot fully is the same type of person who would take the time to completely fill out a survey and share personal information. Furthermore, only 28% of the survey responders were those who partially interacted with the robot. As the robot found people randomly, and we did not specifically recruit people for the test, our data was unbiased. This conclusion is supported by Brickmore et. al. who found that extroverts tended to respond better to agents who leveraged social language [21].

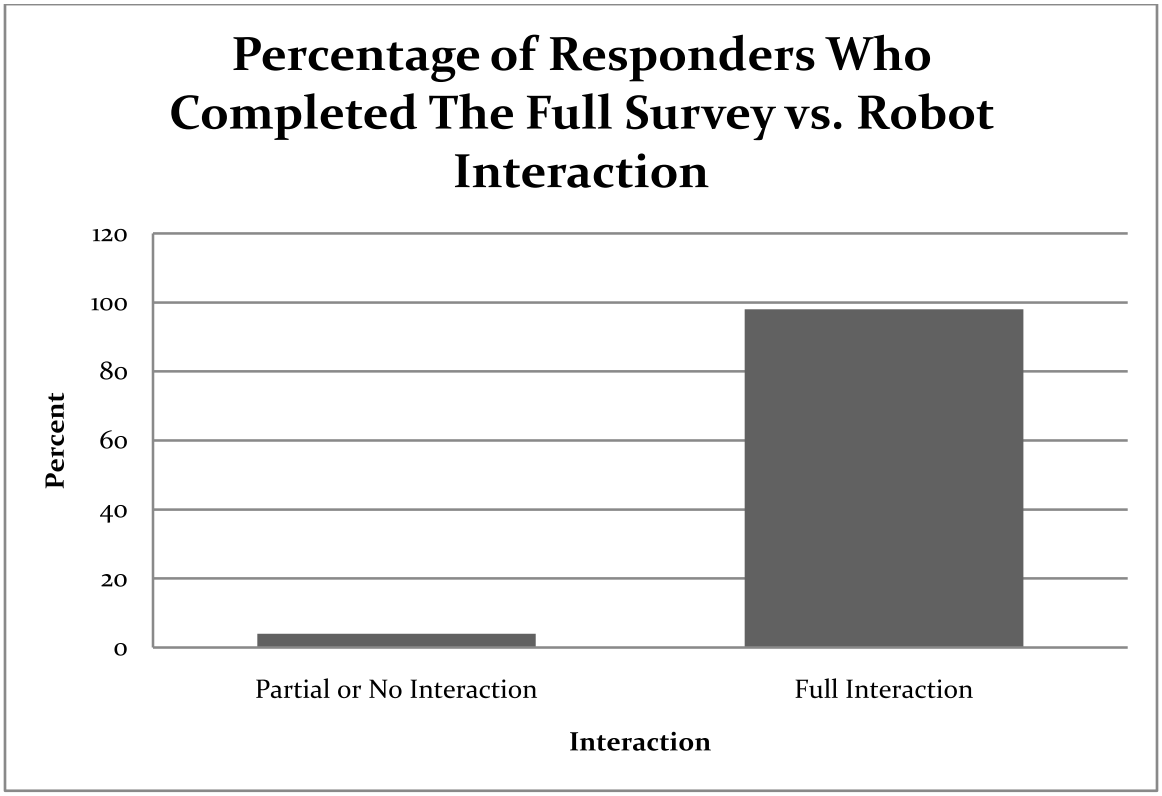


Figure 5.19: Percentage of Responders Who Listed Complete Personal Information vs. Robot Interaction.

Another interesting trend emerged, users who did not complete the full interaction also tended to mistreat the robot. We measured how mistreated the robot was by using simple accelerometer data. While users who reached the full interaction stage reached an average maximum G force of 2.4, those who did not complete the interaction fully reached an average maximum of 4.9 Gs, over twice as much (see Figure 5.20). While 2.4 Gs is within the limit of normal acceleration of the robot associated with handling a force of 4.9 Gs indicate rough handling and a few interactions where the robot was placed back on the ground roughly. Figure 5.21 and Figure 5.22 show histograms of the different types of interactions vs. maximum recorded G force. There is a clear split between the interaction types. This data could be used to sense a bad interaction for either automatic editing or for the robot to take action and try to evade this type of person.

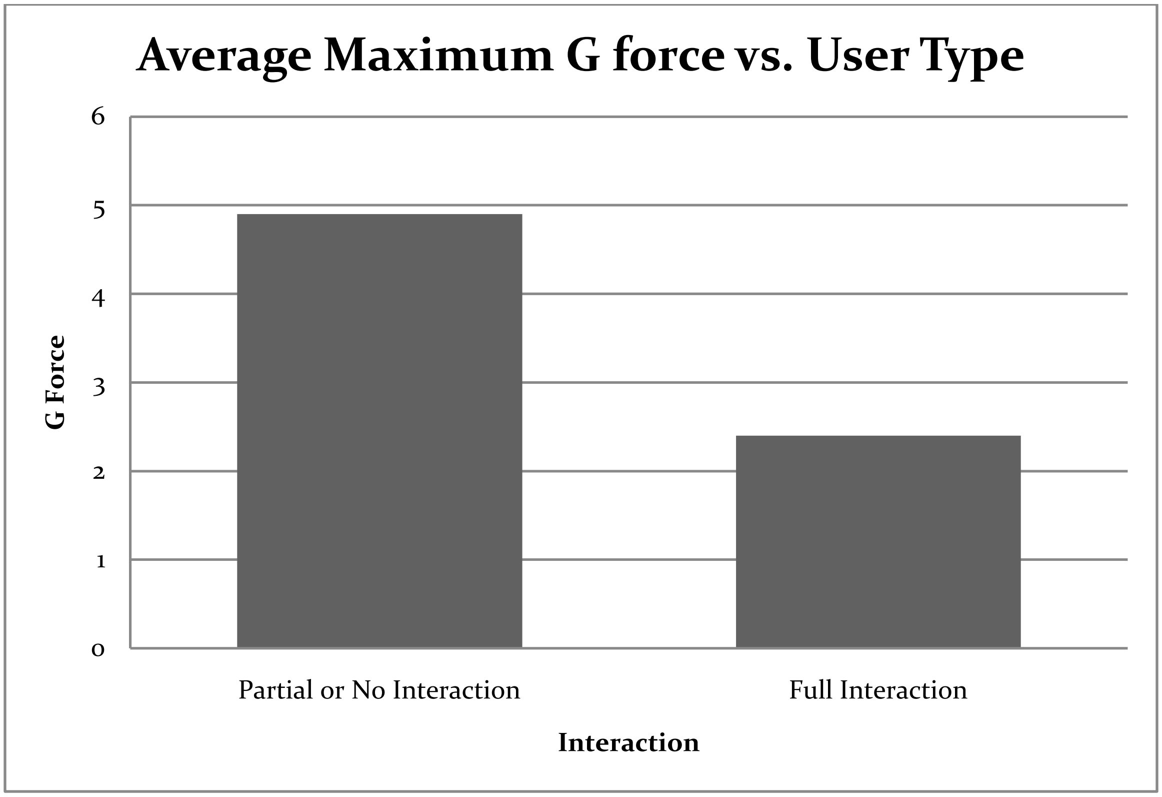


Figure 5.20: Average Maximum G force vs. User Type

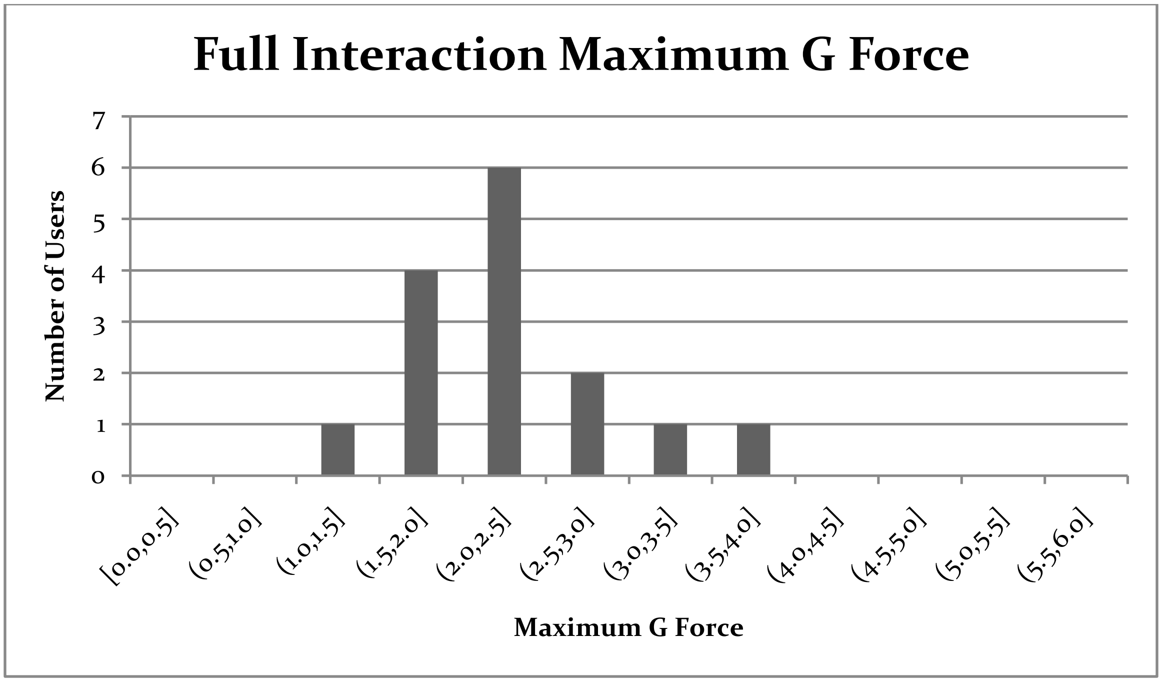


Figure 5.21: Maximum G force for full interaction users.

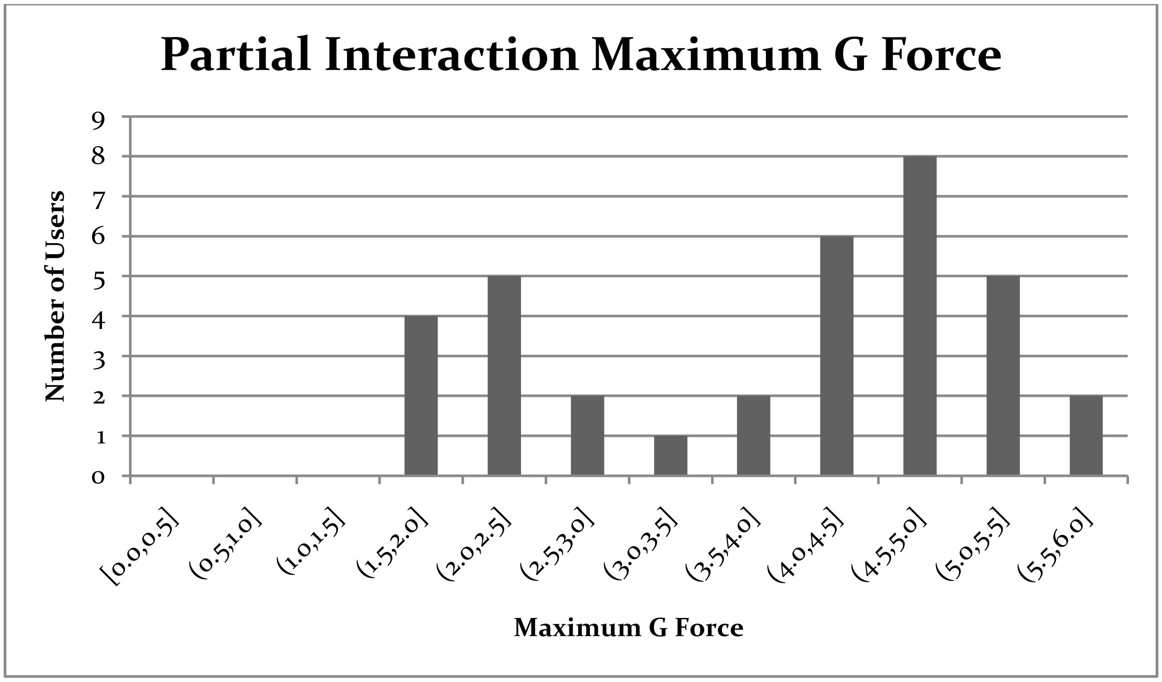


Figure 5.22: Maximum G force for partial interaction users.

Chapter 6

Conclusions and Future Work

6.1 Summary

In this thesis we presented a method for the creation of a physical agent to facilitate interaction and documentary gathering within a ubiquitous media framework as a method for studying human-dependent systems. We built a robot and sent it out to autonomously capture stories about its environment. The robot had a specific story capture goal and leveraged humans to attain that goal. The robot gathered a 1st person view of stories unfolding in real life. We evaluated this agent by way of determining "complete" vs. "incomplete" interactions. "Complete" interactions were those that generated viable and interesting videos, which could be edited together into a larger narrative. It was found that 30% of the interactions captured were "complete" interactions. Results suggested that changes in the system would only produce incrementally more "complete" interactions, as external factors like natural bias or busyness of the user come into play. The types of users who encountered the robot were fairly polar; either they wanted to interact or did not - very few partial interactions went on for more than 1 minute. It was also determined that this type of limited-interaction system is best suited for short-term interactions. At the end of the study, a movie was produced from the videos captured, proving that they were viable for story-making.

6.2 Discussion

We propose the following approaches for building a similar system in the future.

1. Keep the interaction as interesting as possible. Users are more likely to share stories if the agent also shares back with the user.
2. Anthropomorphize the system. Create a connection with the person you are trying to leverage. Have the agent offer them something in exchange for their interaction. Make the agent seem like it needs the user.
3. Be transparent with the purpose of the agent. Users are prone to be skeptical about a system that needs them but does not tell them why.
4. Try not to be annoying. There is a fine line between attempting to capture a user and annoying them; the system is most effective just before that line.
5. Look good. The agent should look the part. It is as important a consideration as the technology inside.

6.3 Future Work

Some of the proposed tests of the system were not able to be executed. These include investigating a gaming type interaction, where the user participates in a game in exchange for interaction. Interactions involving social ties should also be investigated further, including using those ties to leverage a network of connected people. More study needs to be conducted on how to optimize the various aspects of the robot to custom fit a particular location and interaction scenario, as there are many variables to change.

Besides developing the agent for interaction, the ubiquitous network it was tied to needs to be better outfit to this particular application. A denser network of sensors is needed, along with the ability for the nodes to visually track the robot with its camera. This would allow more of the interactions to be captured from a 3rd person view. Future investigations could include deploying several agents into an area at once. The agents could capture interactions from several angles and increase the probability of human acquisition. While it was planned for our system, WiFi tracking was not yet implemented in our building complex. With this type of tracking system in place, the robot could find its position anywhere in the lab and adjust its interactions accordingly. For example, it could be quieter in private places and know where it was being moved to. A system for automatically editing the videos the robot capture could be built (much like the discussed Laibowitz's SPINNER system [1]) since the robot carries sensors and logs activity.

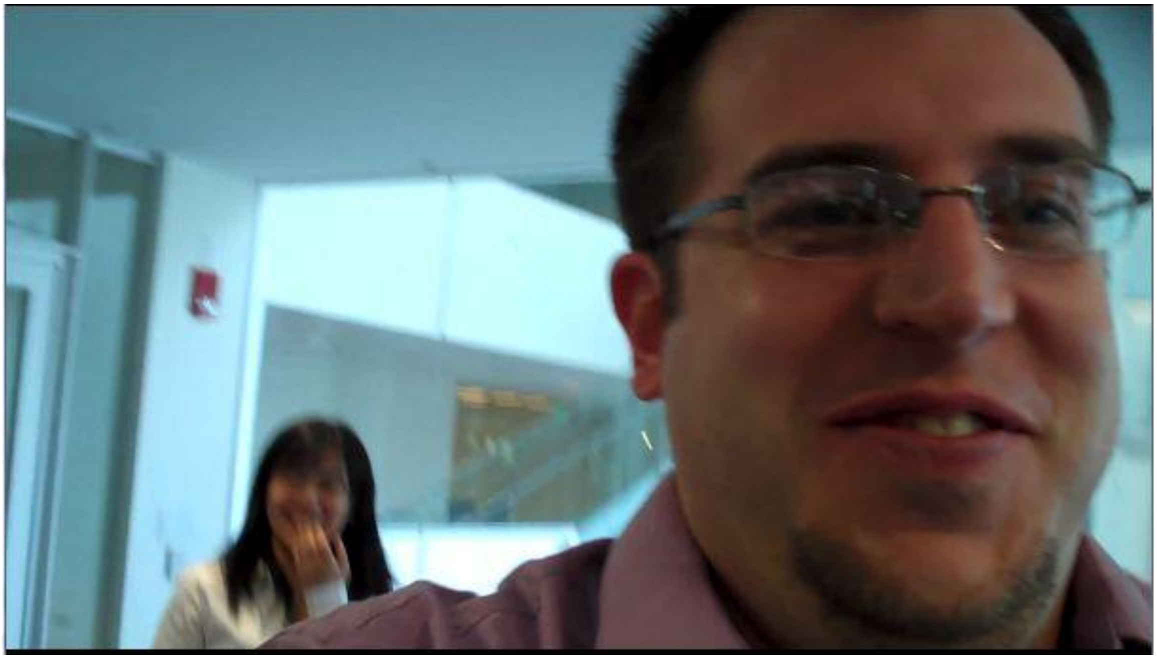
One problem encountered (even with careful instructions to the user and position sensing) was that of properly placing the user in the video frame. A few videos had the tops of user's heads cut off along with other compositional problems. Future systems could track faces and move the camera to position the user properly in the frame. However, upon reviewing footage, badly composed shots occurred infrequently and may not be worth the extra complexity to fix.

6.4 Concluding Remarks

For a system to leverage a human, it first must respect them. This system attempted to exhibit a human touch through the way it engaged users. Boxie represents a link between system and person, which we hope will be used to augment cold systems with some humanity in the future.

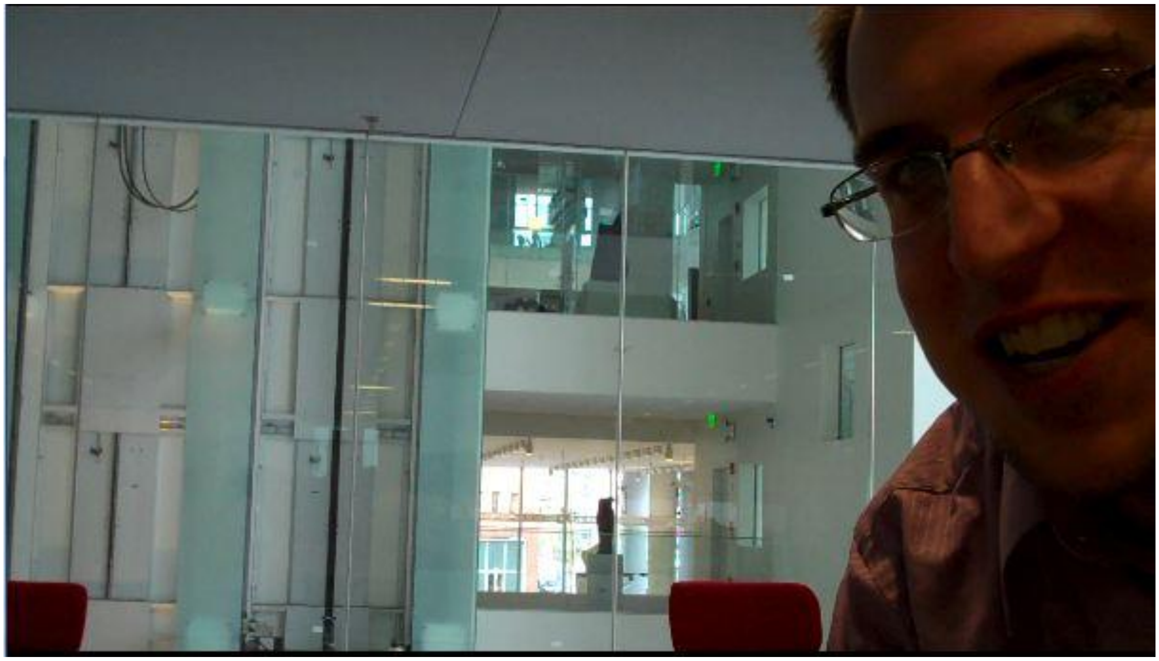
Appendix A: Annotated Video Stills

0:04
A user discovers the robot and presses the green button signifying that they would like to be in the robot's movie.



0:32
The user picks up the robot, moving it to a desk.

0:52
The user places the robot on a desk and continues interaction.

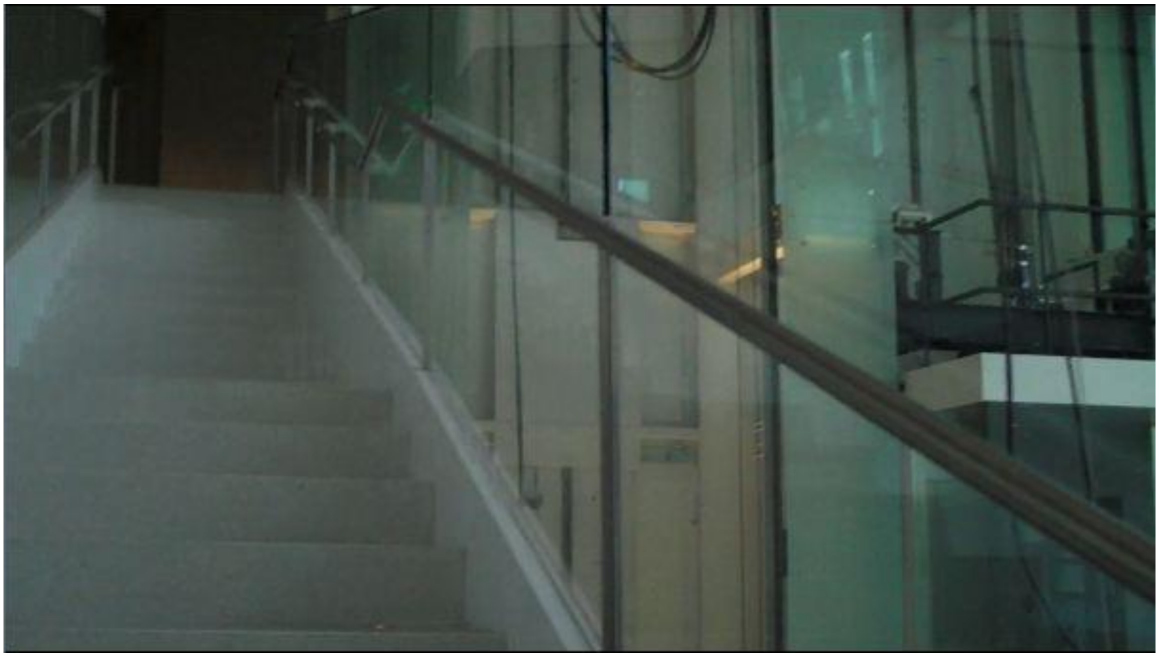


1:09
The user tells the robot their name.

1:12
The user introduces the robot to their friend.

1:24
The user shows the robot the view from their office.

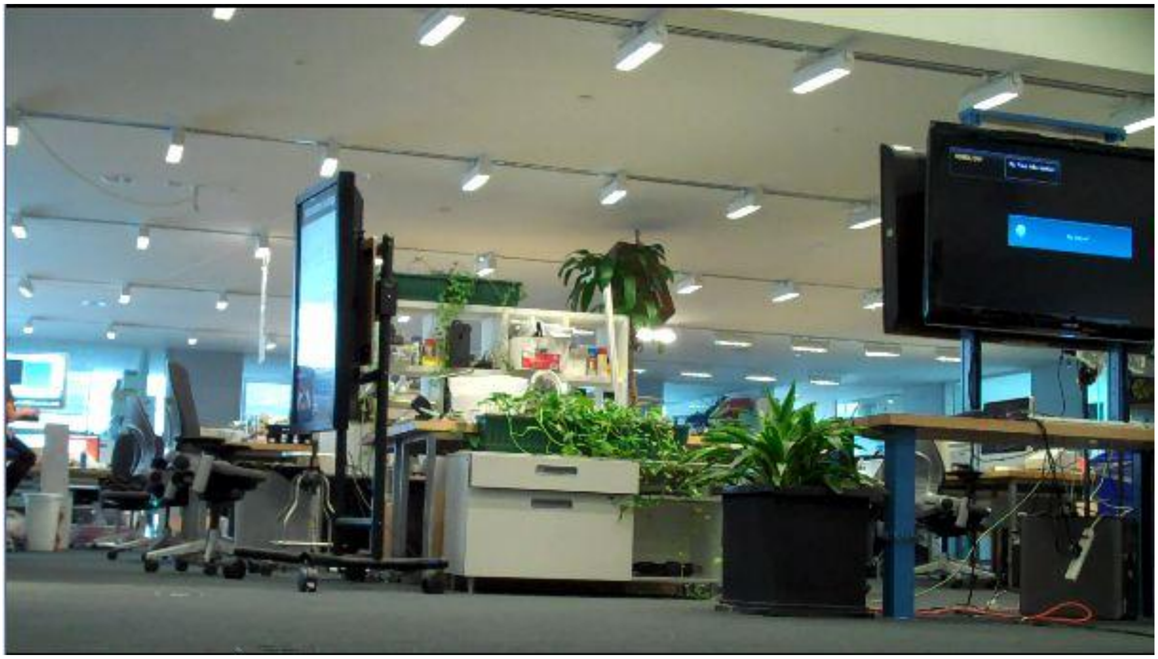
1:48
The user picks up the robot and shows it around their office.



2:05
The user brings the robot up a flight of stairs to show it another location.

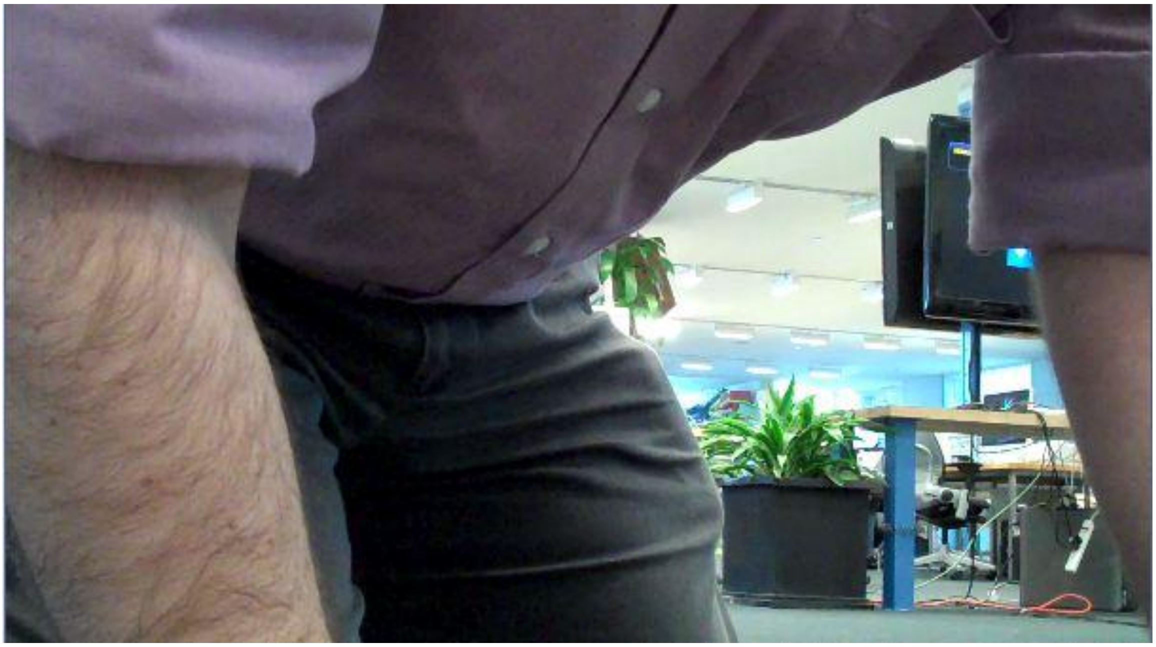
2:21
The user explains this area and the ping pong game to the robot.

2:48
The user shows the robot to a friend and tells the robot about them.



3:08
The user puts the robot down and tries to persuade their friend to interact with the robot.

3:16
The user dances for the robot.



3:59
The user tells the robot their e-mail address and ends interaction.

Bibliography

[1] M. Laibowitz, "Distributed narrative extraction using wearable and imaging sensor networks," PhD., MIT Media Lab, 2010.

[2] C. Breazeal, "Social interactions in HRI: the robot view," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on,* vol. 34, pp. 181-186, 2004.

[3] C. Breazeal, *Designing Sociable Robots*: MIT Press, 2002.

[4] W. Burgard*, et al.*, "The interactive museum tour-guide robot," presented at the Proceedings of the fifteenth national/tenth conference on Artificial intelligence/Innovative applications of artificial intelligence, Madison, Wisconsin, United States, 1998.

[5] M. Laibowitz and J. Paradiso, "Parasitic mobility for pervasive sensor networks," *Pervasive Computing,* vol. 3468/2005, pp. 255-278, 2005.

[6] K. Kinzer. (2009, *tweenbots*. Available: http://www.tweenbots.com

[7] A. Weiss*, et al.*, "Robots asking for directions: the willingness of passers-by to support robots," presented at the Proceeding of the 5th ACM/IEEE international conference on Human-robot interaction, Osaka, Japan, 2010.

[8] Y. Yamaji*, et al.*, "STB: human-dependent sociable trash box," presented at the Proceeding of the 5th ACM/IEEE international conference on Human-robot interaction, Osaka, Japan, 2010.

[9] C. Vaucelle and T. Jehan, "Dolltalk: a computational toy to enhance children's creativity," presented at the CHI '02 extended abstracts on Human factors in computing systems, Minneapolis, Minnesota, USA, 2002.

[10] Available: http://news.bbc.co.uk/2/hi/science/nature/1898525.stm

[11] D. H. Nguyen*, et al.*, "Encountering SenseCam: personal recording technologies in everyday life," presented at the Proceedings of the 11th international conference on Ubiquitous computing, Orlando, Florida, USA, 2009.

[12] N. Bergström*, et al.*, "Modeling of natural human-robot encounters," School of Computer Science and Engineering, Royal Institute of Technology, Stockholm, Sweden, 2008.

[13] D. Dutton, "Aesthetics and evolutionary psychology," *The Oxford Handbook of Aesthetics,* pp. 693–705, 2003.

[14] N. Angier, "The Cute Factor," in *The New York Times*, ed. New York, 2006.

[15] J. Morreall, "Cuteness," *The British Journal of Aesthetics,* vol. 31, p. 39, 1991.

[16] S. Geldart*, et al.*, "Effects of eye size on adults' aesthetic ratings of faces and 5-month-olds' looking times," *PERCEPTION-LONDON-,* vol. 28, pp. 361-374, 1999.

[17] B. DePaulo, "Nonverbal behavior and self-presentation," *Psychological Bulletin,* vol. 111, pp. 203-243, 1992.

[18] C. L. Bethel and R. R. Murphy, "Affective expression in appearance constrained robots," presented at the Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction, Salt Lake City, Utah, USA, 2006.

[19] S. Satake*, et al.*, "How to approach humans?: strategies for social robots to initiate interaction," presented at the Proceedings of the 4th ACM/IEEE international conference on Human robot interaction, La Jolla, California, USA, 2009.

[20] P. Ekman, *et al.*, "Body movement and voice pitch in deceptive interaction," *Semiotica,* vol. 16, pp. 23-28, 1976.

[21] T. Bickmore and J. Cassell, "Relational agents: a model and implementation of building user trust," presented at the Proceedings of the SIGCHI conference on Human factors in computing systems, Seattle, Washington, United States, 2001.

[22] K. P. Schneider, *Small Talk: Analyzing Phatic Discourse*: Hitzeroth, Marburg,, 1988.

[23] C. Cheepen, *The Predictability of Informal Conversation*. New York: Pinter, 1988.

[24] Wikipedia. (2010. Available: http://en.wikipedia.org/wiki/Corrugated_fiberboard

[25] *Atmel ATMega1280 Datasheet*. Available: http://www.atmel.com/dyn/resources/prod_documents/doc2549.PDF

[26] *MMA7260QT ±1.5g - 6g Three Axis Low-g Micromachined Accelerometer*. Available: http://www.freescale.com/files/sensors/doc/data_sheet/MMA7260QT.pdf

[27] *Sharp: GP2Y0A21YK*. Available: http://www.sharpsma.com/Page.aspx/americas/en/part/GP2Y0A21YK/

[28] *XL-MaxSonar-EZ0 datasheet*. Available: http://www.maxbotix.com/uploads/MB1260-MB1360_Datasheet.pdf

[29] *GP ePIR Motion Detection Zdots SBC Product Specification*. Available: http://www.zilog.com/docs/devtools/PS0284.pdf

[30] *Flip Cam Specifications*. Available: http://www.theflip.com/en-us/Products/specs.aspx

[31] *RN-131G & RN-131C*. Available: http://www.rovingnetworks.com/documents/rn-131-ds.pdf

[32] N.-W. Gong, "Configurable Dynamic Privacy for Pervasive Sensor Networks ", Media Lab, MIT, 2009.

[33] *MP3 Trigger Data Sheet*. Available: http://www.sparkfun.com/datasheets/Widgets/MP3Trigger/MP3TriggerV2-User-Guide-2010-04-16.pdf

[34] *TB6612FNG Datasheet*. Available: http://www.semicon.toshiba.co.jp/docs/datasheet/en/LinearIC/TB6612FNG_en_datasheet_080509.pdf

[35] J. Cassell and T. Bickmore, "Negotiated Collusion: Modeling Social Language and its Relationship Effects in Intelligent Agents," *User Modeling and User-Adapted Interaction,* vol. 13, pp. 89-132, 2003.