GUARDED MOTION AND REFLEXIVE BEHAVIORS FOR THE SURVIVOR BUDDY ROBOT

An Undergraduate Research Scholars Thesis

by

OSRIC NAGLE

Submitted to the LAUNCH: Undergraduate Research office at
Texas A&M University
in partial fulfillment of requirements for the designation as an

UNDERGRADUATE RESEARCH SCHOLAR

Approved by
Faculty Research Advisor: Dr. Robin Murphy

May 2022

Major: Computer Science – Computer Engineering

RESEARCH COMPLIANCE CERTIFICATION

Research activities involving the use of human subjects, vertebrate animals, and/or biohazards must be reviewed and approved by the appropriate Texas A&M University regulatory research committee (i.e., IRB, IACUC, IBC) before the activity can commence. This requirement applies to activities conducted at Texas A&M and to activities conducted at non-Texas A&M facilities or institutions. In both cases, students are responsible for working with the relevant Texas A&M research compliance program to ensure and document that all Texas A&M compliance obligations are met before the study begins.

I, Osric Nagle, certify that all research compliance requirements related to this Undergraduate Research Scholars thesis have been addressed with my Research Faculty Advisor prior to the collection of any data used in this final thesis submission.

This project did not require approval from the Texas A&M University Research Compliance & Biosafety office.

TABLE OF CONTENTS

ABSTRACT

Guarded Motion and Reflexive Behaviors for the Survivor Buddy Robot

Osric Nagle
Department of Computer Science and Engineering
Texas A&M University

Research Faculty Advisor: Dr. Robin Murphy
Department of Computer Science and Engineering
Texas A&M University

This paper summarizes hardware and software changes made to the latest version of the Survivor Buddy, specifically regarding the addition of motor burnout prevention systems and the implementation of behavior tracking and mimicking functionality. The Survivor Buddy is an inexpensive and compact robot platform designed to interact with people as a social actor. Its main applications include search and rescue operations and telemedicine, with potential applications in other areas. Currently, robots in these areas are either very expensive, very large, or do not engage with people as social actors. The updates made to the Survivor Buddy outlined in this paper are designed to increase its ability to accurately model human behavior while allowing it to detect when its behavior-based movements are blocked. To create these changes, functionality from existing libraries was combined with data processing and classification algorithms. The implementation of the reflexive system created a system response time range of 0.5-1 seconds, allowing the robot to quickly detect obstacles. These changes demonstrate the efficacy of techniques designed to support inexpensive hardware, thereby providing proof of the practical feasibility of low-cost and functional social robot platforms.

ACKNOWLEDGEMENTS

Contributors

I would like to thank my faculty advisor, Dr. Robin Murphy, for her guidance and support throughout the course of this research. She has done an excellent job of not only guiding me throughout the course of my research, but of going the extra step to be a mentor. Throughout my time working under her, she has consistently emphasized personal growth and learning as the ultimate objectives of our work and has been supportive and understanding at every turn.

I would like to thank Yashas Salankimatt for his work on the hardware section of the Survivor Buddy, which allowed me to test and implement my work with far fewer hardware issues, as well as for creating the behavior tracking software code discussed in this paper. I would also like to thank him for letting me use his exploded view of the Survivor Buddy hardware (see fig. 3.4).

Thanks also go to my friends and colleagues and the department faculty and staff for making my time at Texas A&M University a great experience.

Finally, thanks to my family for their ceaseless encouragement, tremendous emotional support, and constant love.

The Survivor Buddy exploded view (fig. 3.4), Survivor Buddy hardware updates, and original code for the behavior tracking/mimicking system that were analyzed/used for this paper were provided by Yashas Salankimatt.

All other work conducted for the thesis was completed by the student independently.

Funding Sources

Undergraduate research was supported by internal lab funding.

1. INTRODUCTION

The motivating problem behind the research in this paper is: How can we increase the survival rate of, and provide a higher quality of life to, disaster victims? This is important to victims and rescuers for a number of reasons. The physical/mental condition of victims can be monitored for an extended period of time. This can allow faster post-disaster care once the victim is extricated. If the condition of the trapped victim deteriorates rapidly, their rescue priority from the disaster site can be escalated, which may help prevent short- and long-term damage. These benefits are in the best interest of the rescuers as well, whose goal is to rescue victims with as little damage to them as possible. Additionally, victims are able to communicate with family and friends while under the rubble. This will help prevent strong feelings of isolation and stress caused by waiting for an indeterminate amount of time under rubble, potentially without any human contact.

The specific research question being considered is: how can an affordable and compact disaster robot that enables two-way communication while also functioning as a social actor be created? A robot that satisfies these criteria will be able to crawl through tight spaces to reach victims and provide the aforementioned benefits. By acting as a social actor, the robot can potentially increase the amount of information transferred during two-way communication. More effective communication leads to a potential increase in the quality of remote service/care provided by operators to victims and greater benefits from communicating with loved ones.

An affordable disaster robot can allow more rescue teams, especially more rural and/or lesser funded ones, to have access to these services/technologies in the case of a disaster. They will not have to wait for a robot to be sent in from a larger team, which has the potential to decrease the amount of time victims will remain under rubble.

This paper describes the improvements made to the fifth generation of the Survivor Buddy robot. The paper outlines what changes were made, how they were implemented so the idea can potentially be replicated/built on by others, the benefits achieved by these changes, and the limitations of these changes. All changes made to this robot are novel ideas and implementations built to further the development of the Survivor Buddy. They represent incremental steps towards achieving the Survivor Buddy's ultimate objective of being an an affordable and compact disaster robot that acts as a social actor.

The rest of the paper begins with related work, which begins with an introduction to disaster robotics, describes the Survivor Buddy platform, and summarizes the work done on and with it during previous generations. The next section, implementation, introduces a hardware and software description, then describes hardware and software changes and associated information. Following information about all changes, a description of tests run, and corresponding results is present. After this, there is a discussion about various future development possibilities for the Survivor Buddy platform, a summary of the paper, and additional information.

2. RELATED WORK

2.1 Disaster Robotics Overview

Since the Great Eastern Japan Earthquake, robotic systems have become an increasingly significant part of search and rescue operations and will likely continue to be into the near future [2]. In situations involving difficult-to-reach spaces or few human rescuers, the likelihood of a victim encountering and interacting with a robot is high, which presents new opportunities and challenges. Research has shown that the interaction between humans and computers is fundamentally social, and that victims in stressful situations can be further stressed if robots exhibit non-human like behavior [3]. At a disaster such as a building collapse, it can take responders four to ten hours to safely extricate a trapped victim from the time they are discovered. In order to improve survivability, it is valuable to responders and medical personnel to be able to monitor the victim's condition during this time, as well as comfort the victim and keep them calm. Remote monitoring and communication with the victim requires a means of interaction, preferably including two-way audio and video [4]. In order to better interact with survivor victims for long periods of time and address the aforementioned issues, the Survivor Buddy platform has been created [5]; however, human-like motions of its screen must be programmed at the time of operation by a human operator. Additionally, the platform cannot detect when it is stuck when performing certain motions, creating a risk of motor burnout. By improving the sensing capabilities of movement inhibitors and the personification of the Survivor Buddy's movements, the platform will be able to interact more effectively with victims, thus increasing the survivability and quality of life of trapped victims.

2.2 Survivor Buddy History

Born out of a collaboration between Texas A&M and Stanford University in 2007, the Survivor Buddy was originally designed as a disaster rescue robot that would be used to explore and test various areas of HRI (human-robot interaction); specifically, the Survivor Buddy is designed to explore how trapped people interact with the outside world via a robot. The robot's design, structure, and supporting software changed over generations to support both new technology and new research strategies.

One of the original uses of the Survivor Buddy was evident on an episode of the SciGirls show, where interactions of the prototype robot and middle school girls raised important questions about how to improve not only the Survivor Buddy, but how to design robots such that they interact with humans in optimal, cost-effective ways. While the sample size of the study was not statistically significant, the implications of the questions raised open doors to many possible avenues of further study and innovation. For example, one such question was, "what is the right range and velocity of motion?". Without a more detailed understanding of an optimal range and/or velocity of a robot's motion, there is no guarantee that mechanical designs are interacting with people in the most desirable ways [6]. Another question raised by this experiment was how important verbal cues are compared to non-verbal cues. This question shaped the creation of a text to speech kit for the Survivor Buddy that would allow operators to pass information through the robot using a different method [7]. However, the first prototype had significant flaws that greatly diminished from its perception as an intelligent social actor, ranging from movement limitations to mechanical design flaws [8].

The next generation of the Survivor Buddy design focused on updates to the behavioral aspect of the robot. The design process for affective robots can be prone to error due to the subjective nature of criteria for design approval, which vary greatly from person to person. To reduce the high costs of design associated with this error, a three-step process involving engineers, artists, and animators was created and tested on the Survivor Buddy. By introducing animators, artists, and other believable movement experts into the design process, the robot can be prototyped, adjusted, and created with far fewer physical designs, saving time and money. The second generation of the Survivor Buddy, completed during the same duration of design time as the original prototype, was 50% less expensive, 78% lighter, and up to 700% faster than its predecessor [8].

The Survivor Buddy was also used in work highlighting the presence of heuristics in a sentence that can be used to simulate socially acceptable behavior in an effective and far easier manner than manually coupling sentences and actions. The Survivor Buddy was updated to include functionality that mapped certain predefined areas of a sentence to a fixed set of operator and victim input. For example, the robot would perform a specifically programmed action to indicate confusion if the operator made five or more retypes in the span of 15 or fewer seconds. A corresponding research study performed with these changes to the Survivor Buddy showed the effectiveness of these sentence structure/user input based heuristics, which represents an avenue of future work even in the current Survivor Buddy generation [1].

Further generations were developed in coordination with undergraduate students as part of semester long projects. These updates and redesigns were performed to update the hardware and software of the Survivor Buddy as new technology became available, while integrating information learned from previous generations of the Survivor Buddy.

3. IMPLEMENTATION

This research was carried out on a platform called the Survivor Buddy. This social telepresence robot, currently in its fifth design iteration, is designed primarily for use as a rescue vehicle in disaster situations. Additionally, it can be used in a telemedicine context, allowing an operator to independently control the vehicle to gain additional information about the environment of a patient.

The main goal of the Survivor Buddy is to improve communication between its operator and another person by adopting gestures that mimic human body language during speech. In doing so, the operator's words will be more easily understood by the person, and the person will be more encouraged to share information.

3.1 Hardware Overview

The Survivor Buddy (SB) has two main components – a 4 DOF head with a base and phone mount (see Fig 3.1), and a wheeled platform (also called a "skrode"). The skrode has not yet been built. 2 of these DOF correspond to a rotation of the head around its vertical axis, and a rotation along the z axis, which produces movements like nodding. The other 2 DOF describe how the phone can be tilted or rotated while on the head. The phone held by the SB during the research outlined in this paper is the ZTE Blade A5 2020, but the specific phone can be substituted for another, so long as it has a suitably recent Android distribution and high enough camera resolution to enable video communication.

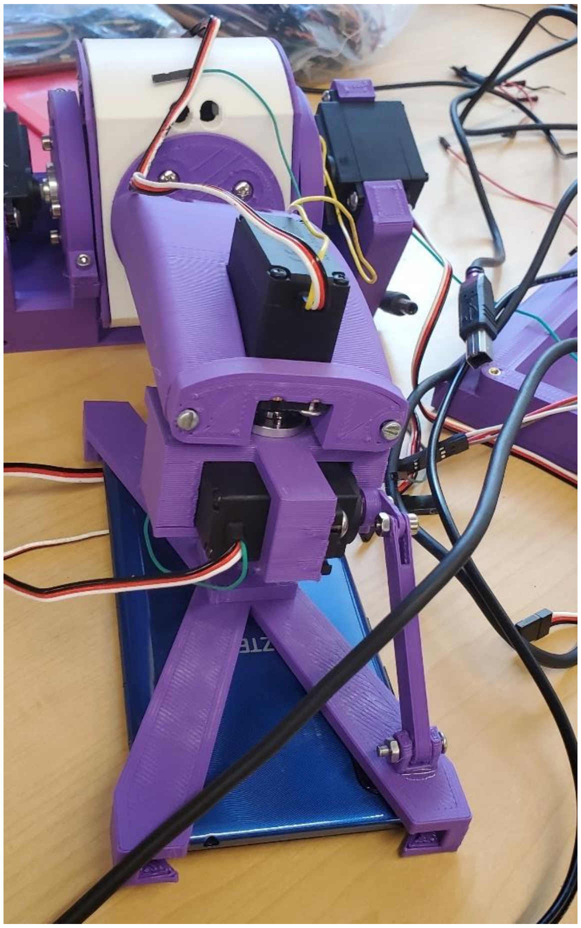

Figure 3.1: SB version 5 head, in construction.

The base of the head is covered by a flat sheet of 3D printed plastic to protect the internal electronics in the base from the phone in case the phone movement exceeds expected rotation limits. These internal electronics consist mainly of an Arduino Uno controller and associated connections to the five servos that drive the movement of the head. Two servos control the rotation of the head along its z axis (which adjusts the height of the phone), while each additional servo controls one discrete DOF. Each of these servos contain a position feedback signal, which is used to perform reflexive movements, as described in a later section. The entirety of the head and base are made of 3D printed plastic, held together with heat set inserts where needed.

3.2 Software and Communication Overview

The software has a client-server relationship, where the applications display possible commands through a GUI (see Fig 3.2). The operator's computer maintains a physical USB connection with the Arduino controller and communicates with the phone via a Wi-Fi signal. When commands are selected in the GUI, the appropriate information is sent to either the Arduino or phone app, which will take the appropriate action. If information is needed, the phone will send it over one of the appropriate servers.

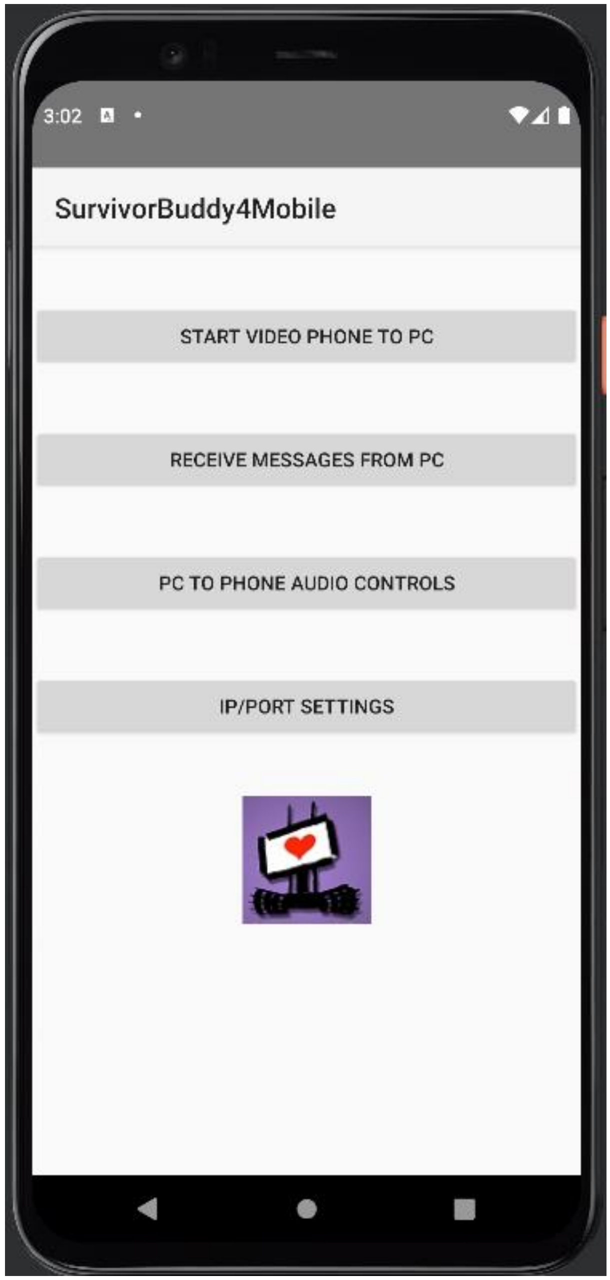

Figure 3.2: SB phone view

The SB interface supports audio, video, message, and phone mirroring capabilities. Each of these can be used independently, or in conjunction, with the others. The audio and video data is captured by the SB phone's microphone and front-facing camera, respectively. At the time of writing, the SB currently contains eight predefined commands (see Fig 3.3), some of which correspond to specific actions designed to mimic human body language during speech. The commands that mimic behavior will manipulate the on-board servos to adjust the position of the head and phone in specific ways to resemble what a person may do when speaking. These commands are:

- Tilt Head – imitates a questioning gesture
- Nod Head – used an affirmative motion
- Shake Head – Used to signify a negative response
- Open Arm – Moves SB head to upright position
- Close Arm – Moves SB head to resting (lowered) position
- Landscape – Changes phone orientation to landscape mode
- Portrait – Changes phones orientation to portrait mode
- Shutdown – Moves SB head to resting position and shuts off communication

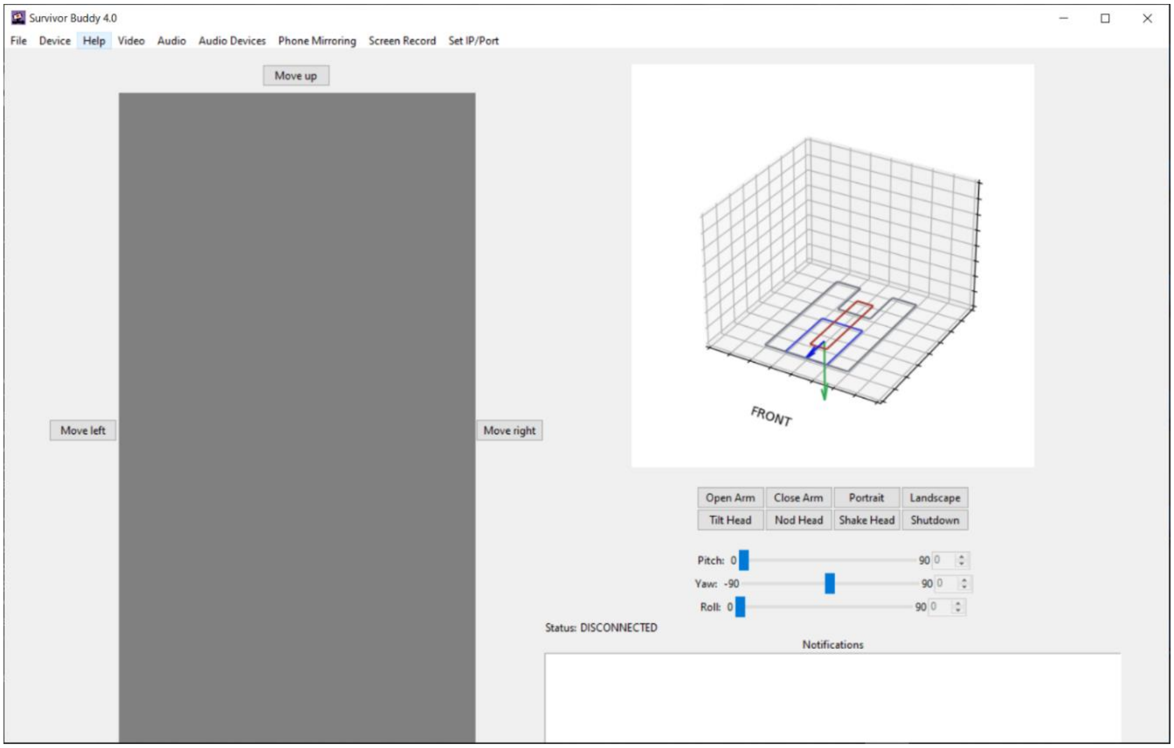

Figure 3.3: SB operator view with eight predefined commands.

3.3 Hardware Changes

Many changes were made to the structure of the Survivor Buddy. All of these changes were made by Yashas Salankimatt, a fellow undergraduate student researcher in the Disaster Robotics Lab. The 3D printed exterior of the robot was redesigned to support a heftier arm, added ball bearings placed in the base to increase the longevity of the base servos, and include a new phone mounting system. The servos were upgraded to the FEETECH FS5115M-FB servos to provide sufficient torque for all movement and generate position feedback that will be used in the reflexive system (see section 3.4). A buck converter decreases the 12V input to 5.7 V, which provides enough power to ensure stable movement of all servos across their entire range of motion. Additionally, to save space, a custom PCB was made around an ELEGOO Nano Board CH 340/ATmega+328P microcontroller, which is housed in a slightly higher electronics enclosure (see Fig 3.4).

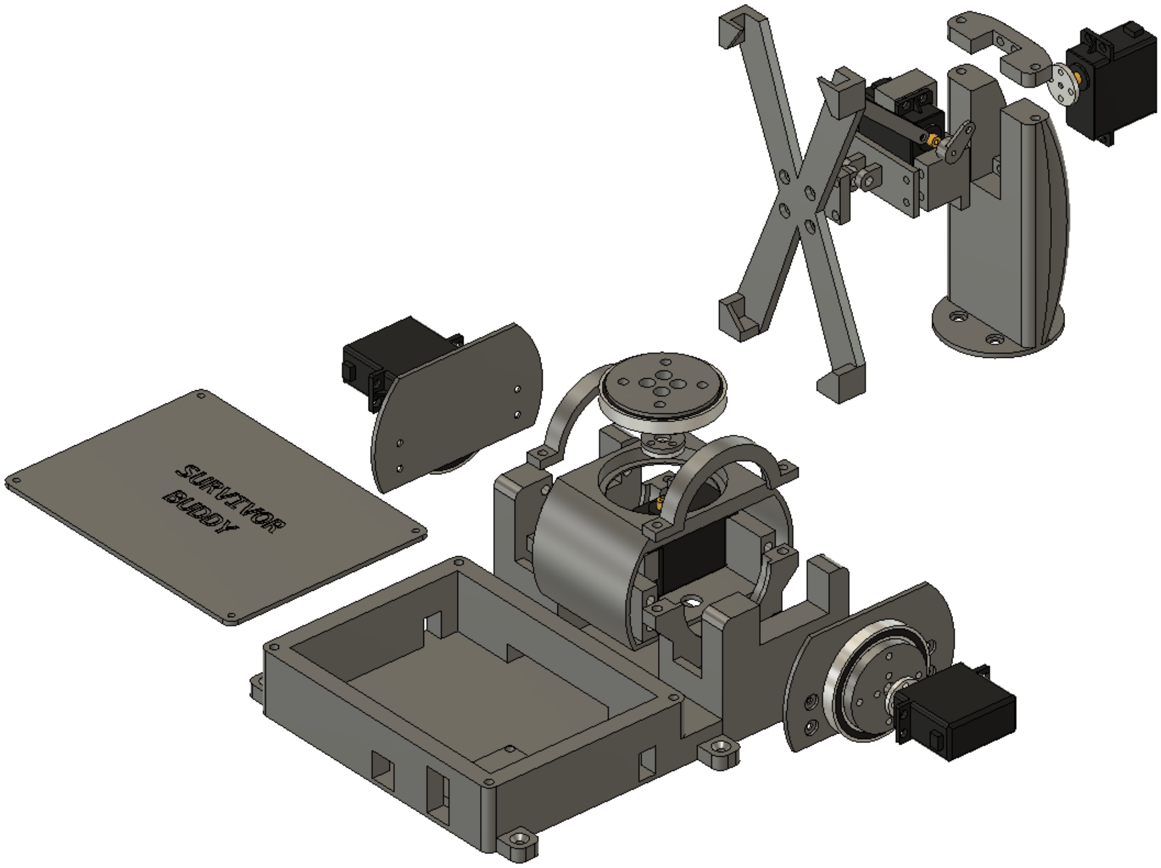

Figure 3.4: Exploded view of the new Survivor Buddy hardware, courtesy of Yashas Salankimatt

3.4 Reflexive System – Intro

One flaw of the previous Survivor Buddy design was the lack of an obstacle detection system when servo movement was impaired. In an actual disaster scenario, there is the possibility that an unseen piece of debris can block a servo from rotating. In the worst case, the operator would perform behaviors without realizing there was any problem, which could cause the servo motors to burn out. Additionally, if used in an educational setting, the Survivor Buddy may have its movement intentionally blocked, and a burnout of a servo due to this would not be ideal. This system should be quick enough to bring the robot to an abrupt stop if any movement impairments are detected.

The reflexive system created for the Survivor Buddy compares actual servo position data to the ideal servo position to detect instances of servo movement impairment. The criteria for a true servo blockage consist of two parts. The first is satisfied if the servo is not in its final position, which is evaluated by comparing the magnitude of the difference between ideal and actual positions to a threshold. The second criterium is satisfied if the servo is not moving, which is evaluated by comparing the magnitude of the difference between actual position readings that occur over a short interval of time with a different threshold. Before the comparison, this difference is smoothed out with an equally weighted average of the current difference value and the previous difference value. When both of these criteria are true, the system will verify they are true twice in a row (to prevent accidents) and then immediately stop the servo that is being blocked.

The reflexive system was implemented on top of the Arduino VarSpeedServo library by modifying the wait() function. The original function only implemented a blocking wait to allow the correct pulse to be sent to a servo. With the revised wait function, the pulse is sent and then a function that checks for movement impairment is continuously called until either that function returns true for movement impairment twice, or the servo reaches its destination.

To support the reflexive system, some functions and values had to be added and kept track of. One such function is a calibration function, which stores the position values at the zero and 180 degree positions of each servo. This information is used in the reflexive system to accurately map the position value information from each servo to a servo pulse width (between 544 and 2400) , so the reflexive system can directly compare ideal and measured servo values. This mapping is slightly different when calibrating the base pair of servos, where both must move at the same time, and must move in opposite directions. There is also a more limited range of movement allowed for the base pair of servos (zero to 135 degrees) due to the height of the electronics box enclosure. Because of this, the mapping of measured servo position value to servo pulse width occurs between a smaller range for the base servos.

3.5 Reflexive System – Experiments/Thresholds

As mentioned above, the criteria for determining whether a servo's movement is artificially stopped are determined based on threshold values. Because only servo feedback position information is used as input for the reflexive system, the reliability of the servo position had to be evaluated. To do so, the position values from a servo were continuously printed to console output while the servo was slowly moved from a resting position throughout its range of motion. The movement was stopped at various points to see how far the position values fluctuated from each other. With this method, it was found that each point was, at most, 3 units of position off of the average for any given servo placement. While present, this represents only a very slight issue, due to the average servo having a range of around 660 units of position to represent 180 degrees of motion; therefore, at worst, each servo's position measurement reading is less than 1 degree off its expected (average) value at any point in the servo's range of movement. Because the reflexive system does not need to be extremely precise, the limited fluctuation in the servo measurement was further reduced in importance.

Initial testing using the servo movement impairment criteria described earlier resulted in the discovery of a tradeoff that would define the performance of the reflexive system. If the threshold values were set relatively high, the system would not flag any false positives (detecting servo movement impairments where there were none) but would not detect true movement impairments. Conversely, creating stricter thresholds would catch all true movement impairments, but would frequently generate false positives. Typically, with strict thresholds, the false positives would occur closer to the beginning or end of a servo movement. This is because these parts of the servo movement tended to satisfy the two servo movement impairment criteria when thresholds were set to be restrictive.

In order to make the tradeoff as acceptable as possible, the false positives needed to be reduced to almost zero, while being able to catch as many true movement impairments as possible. In addition, the reflexive system had to make a decision as quickly as possible, so any changes to the reflexive system could not add too much additional time. To do this, there were a number of combinations of various parameters in the reflexive system that could be modified. The two threshold values to detect each criterium for movement impairment, along with the number of measured position values (which could be averaged before being compared to the ideal position value), time interval between measured values, consideration of previous data points, and amount of successive positives to reach a true classification of movement impairment were all considered as parameters that could alter the tradeoff between false positives and actual movement impairment detection. Due to the continuous nature of many of these parameters, and the correspondingly large number of possibilities of parameter combinations that could be changed, it was impossible to experimentally explore the entire space of parameter possibilities. However, a number of combinations were tried out and tested.

The method of testing parameter changes involved running the adjusted reflexive system through a testing program that moved from zero to 90 degrees repeatedly. The rate of false positives and true detections per servo movement over this interval ranged was determined from the collected data. If the rate of false positives per movement was less than 0.2 and the rate of true detections was 0.9, the system was deemed acceptable enough as an intermediate adjustment. After testing, a single parameter was tuned until either one of these criteria was satisfied, or it was determined that changing a different parameter would produce more desirable results. After rounds of testing and parameter changes, it was determined that both thresholds having a value of 50 with an interval of 25 ms between measured position values would substantially decrease the false positive rate while detecting all true positives.

Even with the adjusted threshold values, false positives still existed, which would translate to undesirable motion stoppages to potential future operators. As mentioned above, many of these seemed to occur at the beginning or end of a servo movement. To counter the false positives that occur at the end of a movement, the difference between measured position values was averaged with the previous difference between measured position values. This allowed the small changes in average position values that would occur as the servo slowed to a stop but was not quite finished with movement to average out with the larger difference from an earlier movement check. By implementing this data smoothing technique, the false positive rate dropped to almost zero false positives per servo movement, with only the occasional false positive occurring during movements.

The final adjustment to the reflexive system was put into place in response to the remaining false positives, which occurred as isolated events at seemingly random times during the testing process. In order to remove them, a condition was added that two positive impairment check results would have to occur in order to qualify as a true impairment detection. This improvement decreased the false positive rate to zero in all subsequent tests.

3.6 Reflexive System – Limits

The reflexive system is designed with the assumption that a true obstacle would completely stop the motion of a servo before it reaches its intended destination. However, this assumption does not take into account what forces are exerted on the Survivor Buddy's body when it encounters an obstacle. Due to the relatively high torque exerted by the Survivor Buddy's servos, an interference of movement would cause the Survivor Buddy's body to twist at various joints and/or move the body. If the main body of the robot was not held in place, this would sometimes mean that the rest of the robot would be moved by the reactionary force generated by the obstacle or parts of the 3D printed frame would break. In these scenarios, the position signal from the servo would still change, which caused the reflexive system to generate a false negative. This scenario was observed specifically when the Survivor Buddy head was close to the zero degree position and a rotation of the head was blocked, but will likely occur with other position and movement blockage combinations as well.

The reflexive system designed for the Survivor Buddy, with all parameter adjustments, was tested using a specific set of hardware. The specific numbers obtained this from testing are therefore specific to the type of servo, and specifically the five servos that are part of the current Survivor Buddy. As a result, there is a non-zero chance that different servos of the same brand and listed specifications, but that have slightly different internal tolerances, will not work as well with the reflexive system. There is also a high chance that a different servo with similar load torque specifications may not work well with this reflexive system.

The first criterium for detecting movement impairments is seeing whether a measured position value from a servo is within an acceptable threshold of distance away from an ideal position value. Therefore, any servo movement impairments that occur within this threshold of distance between actual and ideal positions will not be picked up by the reflexive system. This is deemed acceptable because any servo movement impairments that occur within this distance threshold would not do much to affect the appearances of any predefined behaviors.

Additionally, these servo movement impairments would not cause much servo burnout, since the movement of the servo at that stage is almost complete.

3.7 Behavior Tracking/Mimicry

Currently, the Survivor Buddy uses predefined behaviors to communicate more effectively with disaster victims, patients, or anyone else who comes into contact with the device. Using these behaviors and accurately triggering them at the appropriate places in conversation can be difficult for the operator, as they have to focus on more than just their conversation with the other person. Additionally, these behaviors represent only a very small subsection of the larger group of conscious and unconscious behaviors people exhibit when they communicate.

To automate the movements of the Survivor Buddy to imitate human behaviors, as well as perform a much wider range of behaviors, a behavior tracking and mimicking system was created by Yashas Salankimatt. This system uses a webcam and OpenCV libraries to identify facial features and track their movement across the four degrees of freedom the Survivor Buddy currently has. Movements along each axis are sent to the Survivor Buddy servos, which move in a way that mimic the behavior of the operator.

Once this system was created, it was integrated into the existing Survivor Buddy platform. The python behavior tracking code was added to the GUI as a separate option on the toolbar, which, when pressed, triggers a method to calculate servo positions for all servos given the orientation of the face captured in the webcam. The press of the toolbar option also sends a specific character to the Elegoo microcontroller, which triggers the corresponding method that allows the Survivor Buddy to continuously adjust to position information sent via a serial connection. Both the python and Arduino code that control behavior tracking functionality consist of loops; the python method only ends when the escape key is pressed, while the Arduino method ends when a character is sent from the python method that signifies the end of the loop.

3.8 Behavior Tracking/Mimicry – Limits

The behavior tracking system represents a powerful tool to quickly imitate an operator’s body language, thus enabling very human responses of the Survivor Buddy to a trapped victim. However, the system has not been evaluated with human users, so the benefit brought by this functionality is unclear. Additionally, the current iteration of the system does not utilize the previously described reflexive system, nor does it have its own self protection code, so using the behavior tracking mode of the Survivor Buddy runs a risk of damaging the robot if its movement becomes obstructed.

4. EVALUATION

4.1 Reflexive System

The reflexive system's effectiveness was evaluated through tests where each servo or pair of servos controlling a unique degree of freedom was evaluated independent of the other servos. During each test, the calibration sequence would be run, and the servo would continuously alternate between 0 and 90 degrees. Every time the servo began a movement to one of the angles, this information was printed to the console. The status of the servo – stopped or not stopped – was also printed to the console. After a few seconds of unrestricted movement, the Survivor Buddy would be held such that the servo being tested had its movement completely stopped. The time between the announcement of movement to a new angle and a "stopped" movement output was recorded over multiple (at least 10) iterations of movement to a new angle. The result of these tests was that, across all servos/servo pairs, the reflexive system responded within 0.5 – 1 seconds to all movement blockages. The false negative rate depended on the type of blockage being performed, while there were almost no false positives recorded during testing.

It is important to note that the servo had to be held very still to register a movement blockage as such. In the case where the servo still had some slight movement due to movement not being fully impaired, this would sometimes register as a false negative. This is due to an assumption made about how an obstacle interferes with a servo's movement (see section 2.6).

4.2 Behavior Tracking/Mimicking System

The behavior tracking system is designed to mimic an operator's head position/orientation well enough to communicate additional information that may not be explicitly stated. Ultimately, as long as the mimicry of behavior is sufficient enough to convey the intended body language, the system is considered to work as intended. This term is intentionally vague because various methods can be used to achieve the same result, and the desired functionality does not require rigorous testing to specific standards to function as needed. Therefore, the verification of this system has been less formal than that of the reflexive system.

To test this system, the operator would trigger the behavior mimicking system. Once the tracking system was running, the operator would move their head as far as possible in multiple directions to test each of the Survivor Buddy's degree of freedom. One example of a movement used was moving the head from facing left to facing right, and back again. The operator would watch the Survivor Buddy move in response; if it moved in a way clearly similar to the operator's movement, then this was considered a satisfactory solution. The conditions that satisfied clearly similar behavior were a total Survivor Buddy response time that took no less than 10% over or under the time required for the operator to perform the original movement, and no observable, unintentional stops or rapid speedups during the Survivor Buddy's response movement.

5. DISCUSSION

Most of the work described in this research paper focused on improving the SB platform by adding new predefined behaviors, as well as implementing a reflex function to prevent the SB from burning out. These steps are a necessary part of, but only one step towards, the final goals of this research project – to understand more about social interactions between a person and a robot in disaster rescue settings, as well as produce a working model for an affordable rescue robot that can safely deployed into disaster sites. Therefore, there are a number of avenues for future work, including human studies to further understand the interaction of the SB with people in simulated victim scenarios, field testing, and improved behavior responses.

The current reflexive system for the Survivor Buddy represents a solution that does not fully cover all possible situations. While it can respond quickly and accurately in a decent number of cases, testing revealed some combinations of position and servo movement blockages that either damaged the machine due to the large torque exerted by the servos, or moved the robot itself around the obstacle, which was not intended behavior. Therefore, the next generation of the reflexive system would be one that would retain the same level of response time (if not with improvements), while also reducing the edge cases that result in undesirable behavior.

One potential idea for the next generation reflexive system involves using an inverse kinematics model to find the ideal position of each servo. The position signal from each servo would be compared to this ideal value, and if it the two values did not align within a certain period of time, this would reasonably characterize a blocked servo movement. Depending on the length of that period of time, it is very possible this system would reduce both the response time and false negative rate of the reflexive system while maintaining its true positive detection rate.

Currently, some of the behavior-based responses that the SB can perform must be manually triggered by the operator while they are communicating. This represents an increased responsibility of the operator to accurately match the suggested emotions/actions of the SB predefined behaviors with what the operator is communicating to the patient. If the behaviors and speech do not match up, it will likely cause confusion in the patient, which would undermine the reason for implementing social behaviors in the SB in the first place. Therefore, if the SB could activate predefined behaviors at the correct moments in a conversation, it would allow the operator to focus on other essential tasks while maintaining a higher degree of communication with the patient. This technique is being explored with the use of the behavior tracking/mimicking functionality explained in previous sections, but this has not yet been tested.

Previous research in this area has highlighted a promising option for future development through the use of heuristics that link punctuation to head gaze acts.[1] If the operator inputs information via text, or uploads a pre-written script, the SB could theoretically initiate its own predefined behaviors in accordance with what is being said. Another potential option in this area, which would happen even further down the line, is using artificial intelligence to perform semantic analysis on text and expressing a wider variety of behaviors automatically.

The SB has undergone five iterations of development (as of the time of writing of this paper), but all of these have been proof-of-concept methods of an affordable, small rescue robot, designed to figure out areas that need improvement. The updates to the SB, therefore, have been performed based on needs for the SB that were observed in a lab setting. In order to make the SB a potentially viable system for disaster rescue, testing in environments more closely related to field environments is necessary. At the current moment, this is a longer-term goal, since the SB has not yet finished development on some of its more important features, namely the telepresence and reflex aspects.

One specific aspect of field testing that would be important for future SB development would involve conducting studies in simulated disaster environments to better understand how the SB's predefined behaviors affect communication between an operator and a disaster victim. Ultimately, the goal of this research is to improve the quality of communication using a robot by simulating behaviors, but the extent to which communication is improved, and messages are understood by the victim, would be an important area for further effort to be devoted towards. Results from this research can demonstrate which behaviors, if any, are more effective at expression the intended emotion/information, and which help communication. The results can also provide information to show which behaviors still need to be added, and the priority in which they should be implemented.

6. CONCLUSION

6.1 Summary of Work

This paper details improvements made in the design and creation of the Survivor Buddy 5.0, with an emphasis on software changes made. The structure and hardware of the Survivor Buddy underwent a number of improvements, including a 3D printed frame, a custom PCB with an Elegoo microcontroller, and higher torque servos with position feedback signals. The implementation of a reflexive system to protect the Survivor Buddy during obstructed movements represented a software advancement.

This system compares ideal position values to position values obtained from the position feedback on each servo, and using the measured speed of the servo averaged with earlier speed values, classifies a servo's movement as either obstructed or not. Testing on this system showed that it has a response time of 0.5-1 seconds. The false positive rate is very low; however, in some cases, the reflexive system fails to capture what would normally be considered obstacles to movement. Overall, the reflexive system is functional and works in some use cases but would benefit from future development.

6.2 The Bigger Picture

The work described in this paper is one of a series of development steps to create and improve the Survivor Buddy platform, currently in its fifth generation. Ultimately, the reflexive system and behavior tracking/mimicking system were developed with the goal of proving that the Survivor Buddy, a cost-efficient, inexpensive robot can act as a social actor between an operator and other person in disaster, telemedicine, and other scenarios. In this way, the work described in this paper acts as part of a proof of concept that the Survivor Buddy can exist with the aforementioned characteristics.

REFERENCES

[1] R. R. Murphy, J. Gonzales and V. Srinivasan, "Inferring social gaze from conversational structure and timing," 2011 6th ACM/IEEE International Conference on Human-Robot Interaction (HRI), 2011, pp. 209-210, doi: 10.1145/1957656.1957735.

[2] S. Tadokoro, "Challenge of disaster robotics," 2015 34th Chinese Control Conference (CCC), 2015, pp. 21-22, doi: 10.1109/ChiCC.2015.7259606.

[3] Nass, C., Steuer, J., & Tauber, E. R. (1994). Computers are social actors . In Adelson, B., Dumais, S., & Olson, J. (Eds.), CHI'94: Proceedings of the SIGCHI Conference on Human factors in Computing Systems (pp 72–78). Association for Computing Machinery. https://doi.org/10.1145/191666.191703

[4] R. R. Murphy et al., "Interacting with trapped victims using robots," 2013 IEEE International Conference on Technologies for Homeland Security (HST), 2013, pp. 32-37, doi: 10.1109/THS.2013.6698972.

[5] Z. Henkel, N. Rashidi, A. Rice and R. Murphy, "Survivor Buddy: A Social Medium Robot", HRJ'11, pp. 387, 2011.

[6] R. Murphy *et al*., "Survivor Buddy and SciGirls: Affect, outreach, and questions," *2010 5th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2010, pp. 127-128, doi: 10.1109/HRI.2010.5453233.

[7] Z. Henkel, V. Srinivasan, R. R. Murphy, V. Groom and C. Nass, "A toolkit for exploring the role of voice in human-robot interaction," *2011 6th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2011, pp. 255-256.

[8] R. Murphy, A. Rice, N. Rashidi, Z. Henkel and V. Srinivasan, "A multi-disciplinary design process for affective robots: Case study of Survivor Buddy 2.0," *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 701-706, doi: 10.1109/ICRA.2011.5979977.

APPENDIX

Yashas Salankimatt permission statement for Figure 3.4: “I, Yashas Salankimatt, allow Osric Nagle to use this image in his research paper.”