COMPUTATIONALLY LIGHT AUDIO LOCALIZATION FOR ROBOTICS

An Undergraduate Research Scholars Thesis

by

DENNIS BINFORD, BLAKE CHEVALIER, AND ANTON FEDOTOV

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

UNDERGRADUATE RESEARCH SCHOLAR

Approved by
Faculty Research Advisor: Dr. Oscar Moreira-Tamayo

May 2025

Majors: Electrical Engineering

Copyright © 2025. Dennis Binford, Blake Chevalier, and Anton Fedotov.

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.

We, Dennis Binford, Blake Chevalier, and Anton Fedotov, certify that all research compliance requirements related to this Undergraduate Research Scholars thesis have been addressed with our Faculty Research 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

Page

ABSTRACT 1

DEDICATION 3

ACKNOWLEDGMENTS 4

NOMENCLATURE 5

1. INTRODUCTION 6

    1.1 Sound Source Localization 6
    1.2 Analog Frontend 9
    1.3 Applications 12

2. SUBSYSTEM OVERVIEW 14

    2.1 Analog Frontend 14
    2.2 Localization Algorithm 15
    2.3 Robot System 17

3. DESIGN 18

    3.1 Analog Frontend System Design 18
    3.2 Robot System Design 23
    3.3 Algorithm Design 34
    3.4 Integration 38

4. VALIDATION 41

    4.1 Analog Frontend 41
    4.2 Robot Validation 42
    4.3 Microcontroller Validation 44

5. EXPERIMENTATION 46

    5.1 Microphone Enclosure Experiment 46
    5.2 Algorithm Experiment 52

6. RESULTS 56

    6.1 Angle Approximation 56

6.2 Distance Estimation 57

7. CONCLUSION 58

7.1 Conclusion 58

REFERENCES 59

ABSTRACT

Computationally Light Audio Localization for Robotics

Dennis Binford, Blake Chevalier, and Anton Fedotov
Department of Electrical and Computer Engineering
Texas A&M University

Faculty Research Advisor: Dr. Oscar Moreira-Tamayo
Department of Electrical and Computer Engineering
Texas A&M University

Sound source localization (SSL) is a powerful tool for robotic sensory enhancement. The current state-of-the-art algorithms are computationally expensive and lose accuracy when microphones become closer together; these constraints make audition less viable in smaller robots or in robot swarms. Therefore, this paper explores the viability of using rotating microphones to reduce computations and minimize the distance of sensory nodes. This paper demonstrates simulated localization scenarios and design choices taken to reduce computations and enhance accuracy. The proposed rotational audio localization method is compared with angle of arrival (AoA) and distance estimation metrics of algorithms in current literature. In addition to simulations, a robot and hardware system was developed to test the proposed localization method, collect data metrics, and research improvements to the algorithm and robot design. Using the proposed localization method, microphone placement was minimized, the microcontroller sampling frequency is decreased, and the computation requirements are significantly reduced while localization error remains comparable to methods in current literature. Additionally, experimentation was done using the robot system to research two behaviors: the impact of microphone directionality on the method's accuracy, and the effectiveness of different rotational algorithms at various distances. From simulation, robot development, and experimentation, this paper illustrates a creative solution to make sound

localization more efficacious for robotic applications.

DEDICATION

To our families, instructors, and peers who supported us throughout the research process.

ACKNOWLEDGMENTS

Contributors

We would like to thank our faculty advisor Dr. Oscar Moreira-Tamayo, for his guidance and support throughout the course of this research.

Thanks are also extended to our friends and colleagues and the department faculty and staff for making our times at Texas A&M University a great experience.

Funding Sources

Undergraduate research was supported by The Department of Electrical and Computer Engineering at Texas A&M University via the Engineering Honors capstone program.

NOMENCLATURE

ADC	Analog-to-digital converter
Amp	Amplifier
AoA	Angle of Arrival
BPF	Band-pass filter
GBW	Gain-bandwidth product
PCB	Printed circuit board
PWM	Pulse-width modulation
Op-amp	Operational amplifier
RPS	Revolutions per Second
STL	Three-dimensional model format
TDoA	Time Difference of Arrival

1. INTRODUCTION

1.1 Sound Source Localization

1.1.1 Sound Source Localization Theory

Sound source localization is the process of using an array of microphones to estimate the location of an emanating sound source. It has several applications, some of which are in the field of autonomous robotics [1]. Like humans, autonomous robots can identify sound to detect events in their surroundings and act accordingly. In general, artificial systems of sound localization are most commonly done via time difference of arrival (TDoA) techniques. To determine the delay between microphones, the generalized cross-correlation algorithm is used, where the Fourier transform is taken from each microphone signal and applied to the algorithm with a weighting function [2]. With TDoA, the direction of the signal may be derived using the speed of sound and basic geometry [3]. With a large enough microphone array, the beamformer energy technique may be applied where a likelihood is assigned to each possible speaker position.

In recent publications, there have been optimizations to the localization algorithms to enhance their accuracy by adding various weights to the algorithm [2]. It has also become a topic of interest to combine TDoA techniques with angle of arrival techniques [4]. The main caveat with the current research is that the algorithms are computationally expensive, and localization tends to become less accurate for robotics, when the distance from microphone to source is larger than the microphone array [2]. There has been extensive research regarding optimizing localization algorithms for accuracy, but even they tend to require a microphone array that either encompasses the sound source, or an array with many microphones; both solutions are problematic for robotic applications [5]. To circumvent these tradeoffs, we construct a localization method that uses rotating microphones to lighten computations and maintain accuracy.

1.1.2 Sound Source Localization Methods

1.1.2.1 Time Difference of Arrival

Time Difference of Arrival (TDoA) is a common method for audio localization. This method involves using an array of microphones and recording the time of sound arrival at each point. By using a cross correlation algorithm between all microphone pairs, the location may be approximated with geometric estimations. The most common cross correlation algorithm in the generalized cross-correlation algorithm, seen in Equation (1) [5].

$$\tau = \arg\max_{\beta} \left(\int_{\omega} W(\omega) X_i(\omega) \overline{X_j(\omega)} e^{-j\omega\beta} d\omega \right) \tag{1}$$

Where X_i is the Fourier transformed signal at microphone i, X_j Fourier transformed signal at microphone j, and $W(\omega)$ is a weighting function used to tune for accuracy or noise. This method is able to achieve moderate accuracy at the expense of computation requirements and complexity. The accuracy of this method is highest when the distance between microphones is larger than the distance from the source to a sensory node.

1.1.2.2 Beamforming

Beamforming is one of the most common and effective localization methods. By relying on the generalized cross-correlation algorithm, a likelihood value L is assigned to each possible coordinate position using Equation (2).

$$L(x) = \sum_{i<j} \int_{\omega} W(\omega) X_i(\omega) \overline{X_j(\omega)} e^{-j\omega\tau_{ij}(x)} d\omega \tag{2}$$

In practice this method has seen high noise resistance for microphone arrays with relative distances as small as 15 cm [5].

1.1.2.3 Distance Estimation

In many systems, distance estimation is completed first, then the coordinate position is determined from the distance estimation relative to multiple microphones in a microphone array.

Figure 1 illustrates this idea, where a circle representing distance is drawn around each microphone, and the sound source is the point where all the circles intersect [6]. This method is not readily functional for robotics since the microphones are placed around the sound source. For robotics, the entire microphone array should be mounted to the robot, while the sound source may be a variable distance away.

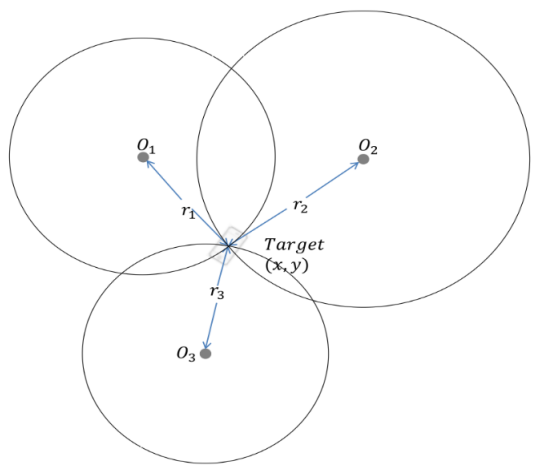

Figure 1: Localization With Given Distances

For localization methods that only obtain the angle of arrival, distance estimation is also required for a full coordinate location. Given the angle of arrival for two microphones, the distance can be estimated by triangulation. Using basic trigonometry and the law of sines, the distance may be determined. Figure 2 demonstrates this graphically.

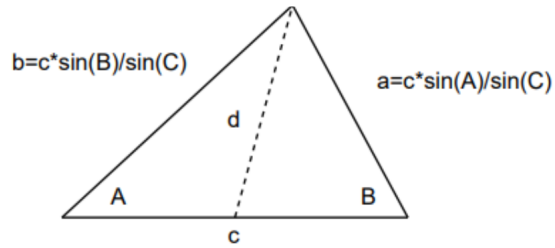

Figure 2: Triangulation

If we assume that the angles A and B are already determined by an angle-of-arrival method, and if we know the distance between the microphones is c, then the remaining triangle values can be computed using Equation (3) [7].

$$\frac{a}{\sin A} = \frac{b}{\sin B} = \frac{c}{\sin C} \tag{3}$$

1.2 Analog Frontend

1.2.1 Propagation

In acoustics, sound waves are mechanical waves that propagate throughout a medium using vibration. These acoustic waves have various properties depending on the medium, sound frequency, and environment. If the sound is propagating through a Newtonian fluid, such as air, then Stokes's law of sound attenuation will describe its behavior.

$$\alpha = \frac{2\eta\omega^2}{3\rho V^3} \tag{4}$$

Stokes's described law may be seen in Equation (4), where η is the viscosity coefficient of the medium, σ is the sound frequency, ρ is the density of the medium, and V is the speed of sound. This equation gives a value for α, the attenuation coefficient. This value may be used to simulate sound attenuation and describes the exponential decreases in amplitude over a distance. Additionally, with a higher frequency sound, the attenuation is naturally steeper [8].

1.2.2 Analog Filtering

Any analog signal, whether that be of sound, light, vibration, or radio wave, can be created by a unique sum of sinusoids of different frequencies and amplitudes. For a certain signal, the frequencies that its characteristic sinusoids occupy is called its spectrum. Filters operate on specific parts of the frequency spectrum, attenuating some parts while leaving others at their original levels. Types of filters include low-pass, high-pass, and band-pass, which all refer to the part of the frequency spectrum they do not attenuate. Another property of a filter is its order, which corresponds to how steeply the transition from the passing region to the attenuation region occurs [9].

Filters for electronic signals can be designed and implemented with analog circuit techniques. Analog filters fit into passive and active classifications. Of these, active filters are the most practical for high-attenuation, higher-order filters. These consist of operational amplifiers, resistors, and capacitors. Unlike passive filters, they require power to function [5].

This paper focuses on an application using an active band-pass filter. The generalized second-order band-pass filter response is written in Equation (5):

$$|H(s)| = A\frac{\frac{\omega_0}{Q}s}{s^2 + \frac{\omega_0}{Q}s + {\omega_0}^2} \tag{5}$$

where s is the complex frequency, A is the low-frequency gain, and Q is the Q factor, a measurement of the steepness of the filter's response.

There exist multiple different active filter topologies, one of which is the multiple-feedback (MFB) topology, which is second-order by nature. A schematic of the generalized topology is illustrated in Figure 3.

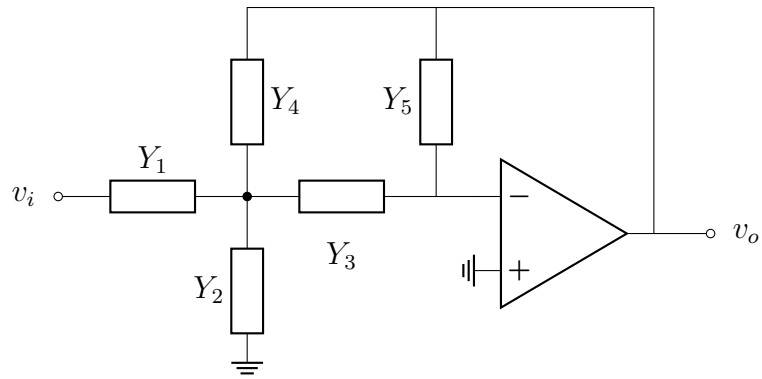

Figure 3: Generalized multiple-feedback active filter topology

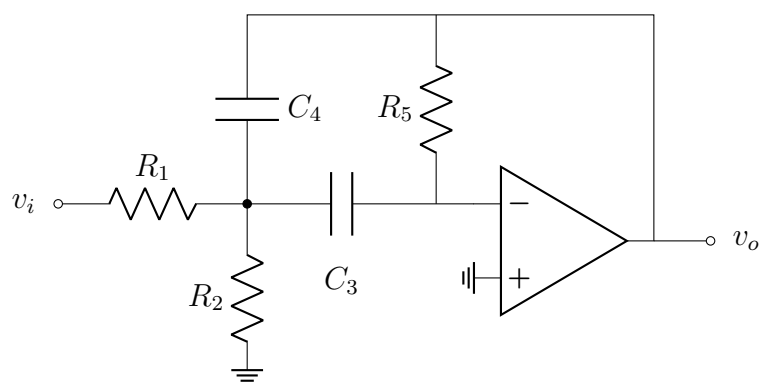

Figure 4: A band-pass multiple-feedback active filter configuration

1.2.3 Sampling and Signal Integrity

To transform a signal such as audio into data usable by a digital system, the signal must be sampled and quantized. Sampling entails breaking up a continuous signal into a discrete and uniform number of sections in the time domain. For a signal to be sampled without a negative impact to signal integrity, the sampling frequency must be at least the Nyquist frequency. The Nyquist frequency f_s is twice the maximum frequency f in the signal as seen in Equation (6).

$$f_s \geq 2f \tag{6}$$

When the Nyquist inequality is violated, the signal loses integrity due to an effect called aliasing. As an effect of aliasing, the signal will be misrepresented by its discrete counterpart [10]. Figure 5 illustrates the effects of aliasing:

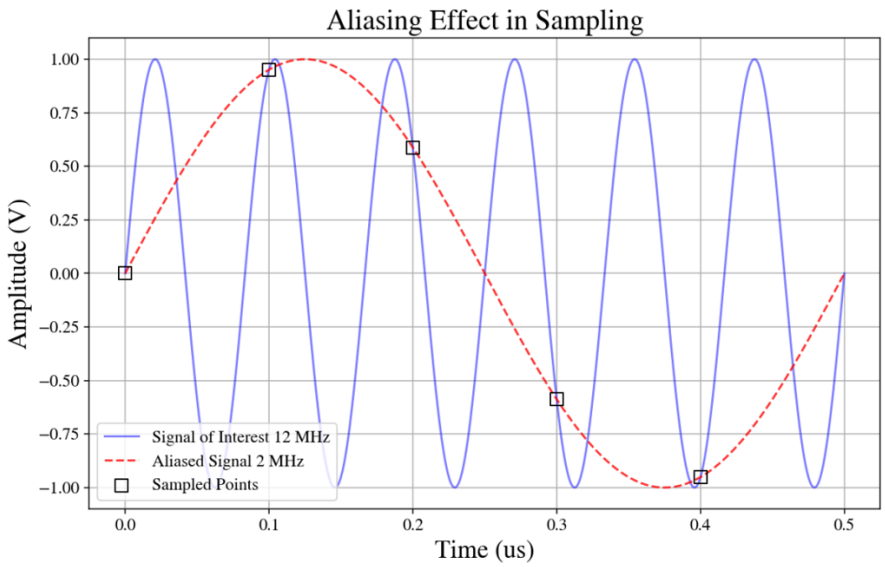

Figure 5: Effects of Nyquist Rate Violation in Sampling

Quantization is also required to quantify the amplitude of the signal as a binary string. To quantize the signal, the amplitude is divided into N equal divisions, where $N = 2^n$, with n being the number of transferrable bits. For each discrete sample, the signal is categorized into its nearest bin.

1.3 Applications

Sound source localization has industrial and commercial applications relating to current technologies. These applications extend to internet of things (IoT), localization for defense, and robotic enhancement. In IoT networks, several sensory devices integrate to perform various tasks in a household or commercial building. Through reduced computations in audio localization, less data will have to be managed by the central processor, allowing for less latency, and a higher

density of devices.

Regarding defense applications, quick and accurate localization of a target is key. For an autonomous system, computational resources may be needed for navigation, communication, surveillance, or other internal computations. By reducing the computational requirements of the targeting system, these technologies may in return be cheaper or allow for more advanced computations in other sensory systems.

In the realm of robotics, the aim is to consistently improve the quality and accuracy of the sensory system while reducing the cost. Complex sensory systems for robotics require numerous real-time computations; by reducing the computational expense of localization, the processing system is freed up to perform computations to improve the accuracy of other sensory nodes. The reduced computation may also be used to cheapen the processing system, where a microcontroller with less storage and RAM will be needed [11].

2. SUBSYSTEM OVERVIEW

2.1 Analog Frontend

An external sound source, which is the localization target, emits an omnidirectional single-frequency sound. It is important to select this frequency to not interfere with other sounds in the expected operation environment. In our case, considering just typical ambient noise, 40 kHz was chosen as the operating frequency. Two highly directional microphones on the robot produce electronic signals representing the sound as it appears to them. It is also important that the microphones are directional, as this property is exploited by the localization system. The signal from each microphone is processed by a series of transformations. Each microphone has its own dedicated signal path, described in the following.

The signal amplitude produced by these microphones is small, and the microphones' output impedance are large, so an amplification block is required to bring the signal to a level that is usable by later processing stages. The amplification factor must be chosen so as to not saturate the amplifier's output or any subsequent block. Therefore, the amplification factor is primarily influenced by the supply voltage powering the active circuitry comprising the audio subsystem, as well as the limits of the input that the audio subsystem's final output feeds into—the microcontroller's ADC. It is desirable to amplify as much as possible within these constraints to maximize the dynamic range.

After amplification, the component of the signal corresponding to the sound source must be separated as much as possible from interference and noise. In the case of a single-frequency sound localization target, a straightforward band-pass filter can be used, centered at the sound source's frequency. This filter will attenuate signal components corresponding to ambient noise outside of the operating frequency, which could otherwise mislead later parts of the localization system. Since the microphones are directional, the signals after the filter block should correspond to the angle of the microphones in relation to the sound source, with the highest amplitude being produced when the microphones point directly at the sound source.

The goal of this system is to require little digital signal processing, and by extension only low-spec digital devices. In accordance with this principle, the speed of the ADC at the interface with the digital processor does not have to be high if it can be helped. If the signal processing from this point were to be done in the digital domain, the signal would first have to pass through an ADC. For a high frequency sound source, sampling this sound for further digital processing requires an ADC which can sample at least double the sound frequency, as exhibited in Equation (6). The next block, the peak detector, eliminates the requirement of sampling at the Nyquist frequency of the signal and allows use of an ADC of a far lower frequency specification by extracting the relevant information contained in the input signal to a low-speed DC signal.

After the filter, the information in the signals relevant to the localization system is only in their amplitudes, which now closely correspond to the angle of each microphone in relation to the sound source. Instead of carrying this information in the amplitude of a relatively high-frequency sinusoid, this information can be represented with a simple voltage. A peak detector does this; it outputs the voltage of the peaks of its input signal. The core components of a peak detector are a rectification diode and an resistor-capacitor network in a low-pass configuration.

In the case of the robot sound localization system, it is reasonable to say that the microphones rotate a speed several orders of magnitude lower than the sound source's operating frequency. In this case, the speed that the microphones rotate at is insignificant to the peak detector, and it only smoothens out fluctuations of the magnitude of the operating frequency, while remaining responsive to changes in its amplitude. This way, the output of the peak detector changes much slower than the sound source's operating frequency, which does not require a fast ADC, making it feasible to use less expensive hardware.

2.2 Localization Algorithm

The localization algorithm finds the peak in the digital signal fed to a microcontroller and the angle the microphone is facing fed from the motor. The two microphones independently find the two data points with the highest peak of the digital signal and use the corresponding angle the microphone was pointing to triangulate the direction of the sound. This direction is marked down

as a specific angle from the robot's current forward direction. The microcontroller then instructs the motors of the robot's wheels to rotate in place in the direction of the sound. The algorithm then conducts a second sweep with each microphone to find the new angle at which the peak is detected for both microphones. Next, the algorithm then triangulates the distance the robot is away from the sound and uses this information to make the robot go that specific distance forward to approach the sound.

This research explores a localization algorithm for calculating a distance d and an angle offset D between an ultrasonic sound source that is perpendicular to the axis line formed by the two microphones and the center of the robot's body. The setup places two microphones at fixed points, designated as A and B, with each microphone oriented to read 0 when facing directly toward the opposite microphone. The positive and negative angles correspond to upward and downward directions relative to the line between the two microphones. The angle formed by the sound source and the microphone line, denoted C, is calculated based on the angles from the sound source to microphones A and B, using the formula $C = 180 - A - B$. The distance c between the two microphones is a known fixed constant that can be set to the measured distance of the two microphones when physically placed together. The objective is to calculate d, the distance from the center of the line formed by the microphones to the ultrasonic sound source and D, the angle that the robot must rotate to point straight towards the ultrasonic sound source.

The algorithm uses trigonometric principles, specifically the laws of sines and cosines, to calculate the distances a and b from each microphone to the sound source, followed by the calculation of d and D, where D indicates the direction of the sound source relative to the microphone line. The code implementation includes functions for converting between degrees and radians, calculating distances and angles using the law of cosines, and determining the angle offset between the microphone line and the direction to the sound source. The process begins with user input for angles at each microphone which are converted to radians for trigonometric calculations. Using the law of sines, the distances a and b are determined, and the law of cosines is then applied to calculate d and the angle offset D.

An analysis of the algorithm's sensitivity to angle measurement inaccuracies demonstrates the critical impact of precise angle input. Even small discrepancies in angles, ranging from 0.1° to 5°, result in significant deviations in the calculated distance and angle offset, with errors ranging up to 11.49 percent in distance and 43.9 percent in angle offset for a 5° error. This sensitivity highlights the importance of accuracy in angle measurements for reliable localization results, as even minor inaccuracies can lead to substantial errors in determining the sound source's position. The findings highlight the need for precise measurement methods or alternative computational approaches when greater localization accuracy is required.

This localization algorithm differs significantly from traditional TDoA approaches. While TDoA relies on a fixed array of microphones and calculates the time delays between sound arrivals at different microphones, the rotational method uses a pair of rotating microphones to measure the angle of arrival directly. This method reduces the computational complexity, as the algorithm does not need to perform cross-correlation calculations or handle large arrays of microphone data. Furthermore, the rotational method requires only two microphones that can be in close proximity, making it more suitable for small robots or systems with limited processing power.

2.3 Robot System

The main objective of the robot system is to develop a platform with limited area to mount the localization system. Additionally, the robot communicates with the localization system to navigate towards the sound source. The robot will also be responsible for distributing power to the wheels, microcontroller, and to the stepper motors used for rotating the microphones.

The robot will have two wheels controlled by servo motors for movement and two free wheels for balance. There will be an additional two stepper motors for the two microphones mounted on top. The robot will have the sound transmission system circuity on a PCB that feeds into a cheap and memory light microcontroller for the digital signal located on the top. The robot will control the rotation of the microphones for localization sweeps and the wheels for movement toward the sound source.

3. DESIGN

3.1 Analog Frontend System Design

The operating frequency of the sound source was chosen to be 40 kHz due to the expectation of less ambient noise in this range, and to be out of the range of human hearing. The entire analog frontend system as described in Chapter 2 was implemented on a PCB as a daughterboard designed to attach to the Arduino Uno R3. This microcontroller's operating voltage is 5 V, so for simplicity, the whole analog frontend was designed to operate at this voltage as well. Power was delivered to the frontend and microcontroller through a fixed-output 5 V linear regulator, an LM7805 [12].

Given the measurement of the peak raw amplitude from the microphones when facing the sound source at a short distance, the amplification factor was chosen to be 35 to maximize use of the available range limited by the operating voltage. The amplifier was constructed using a inverting op-amp configuration using an LM6132 op-amp as shown in Figure 6. A level shifter is also added before the amplifier in the design to center the frequency at the midpoint of the operating voltage, 2.5 V.

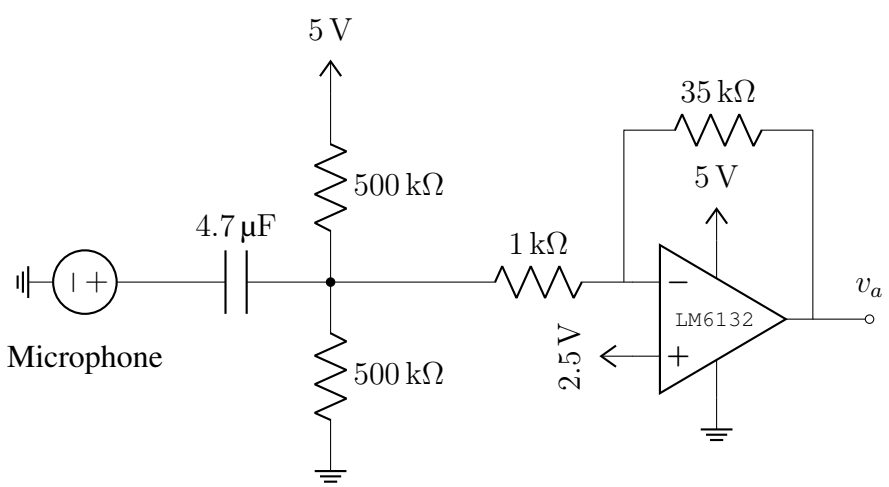

Figure 6: Amplifier as designed

The LM6132 was selected due to its availability of multiple units on one chip, its rail-to-rail output, and its low power consumption. The rail-to-rail property is especially important for being able to maximize use of the limited operating voltage range. The combination of these attractive properties made the LM6132 the preferred choice for all op-amps where possible [13].

A fourth-order filter was designed to adequately attenuate frequencies outside of a narrow range around the operating frequency of 40 kHz. The filter was created by combining two band-pass multiple feedback filters in series, with one centered slightly below the operating frequency and one centered slightly above. The TI Filter Design Tool* was used to assist with designing the filter described in Table 1. The design criteria were chosen to follow the Bessel response because it requires lower-performance op-amps compared to the other target responses. [14]

Table 1: Band-pass filter parameters as designed

	f_0	Q	Required GBW
Stage 1	$39.085\,\mathrm{kHz}$	18.05	$82.447\,\mathrm{MHz}$
Stage 2	$40.723\,\mathrm{kHz}$	18.295	$85.115\,\mathrm{MHz}$

As implemented, the filters use the OPA4354 op-amp chip [15]. This is an array of four op-amps with GBW suited to the requirements of the filter design. Since a total of two filters are required for the whole design, one for each microphone channel, this chip is especially well-fitting, reducing implementation costs at the same time. These op-amps are also rail-to-rail, meaning they can output along the full range of the limited supply voltage. The fully-designed filter is illustrated in Figure 7 and its simulated response is shown in Figure 8.

*https://webench.ti.com/filter-design-tool/

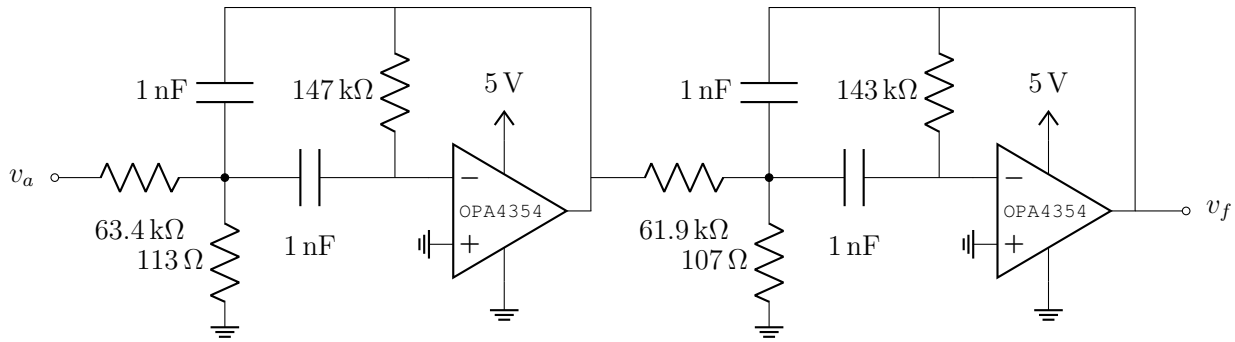

Figure 7: Band-pass filter as designed

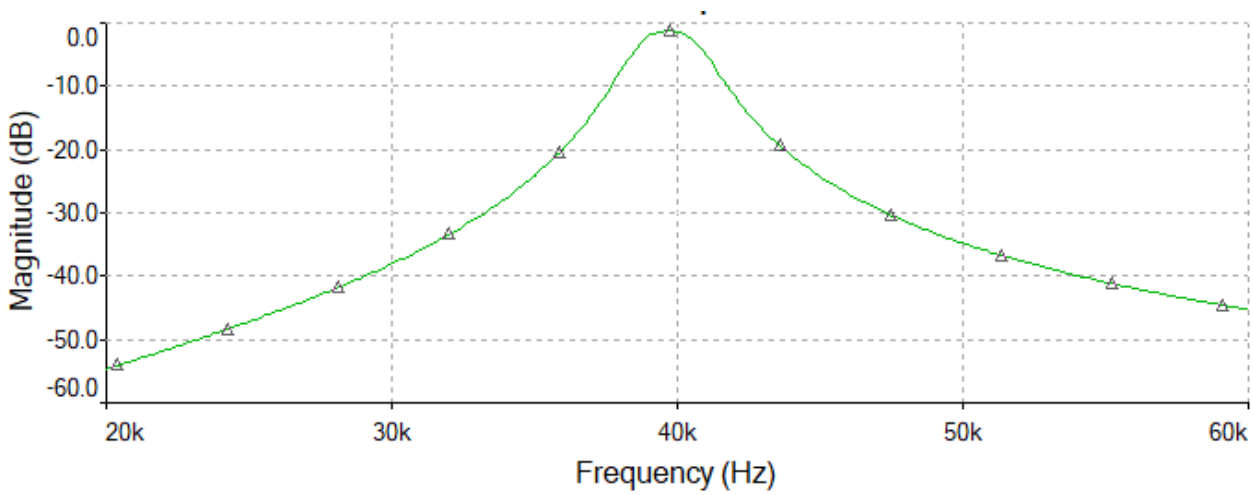

Figure 8: Simulated response of band-pass filter as designed

The Arduino Uno R3 comes with several built-in analog-to-digital converters, however, they are relatively slow, and have sampling rates far below the Nyquist frequency of 40 kHz. This negative effect requires the use of peak detection circuitry for the microcontroller to receive usable information. A robust peak detector as implemented is illustrated in Figure 9. The core components of the peak detection circuitry are the diode D_2, the 680 nF capacitor, and the 100 kΩ resistor. The capacitor provides memory to the circuit holding the peak value, charging up before the signal peaks and holding its voltage level after the peak. D_2 provides isolation between the

input and the output preventing the capacitor from discharging towards the input when the signal is falling, and ensuring that the only discharge path for the capacitor is through the resistor. The discharge through the resistor allows for the capacitor voltage to fall should the amplitude decrease from a previously larger value. The product RC be chosen to be much larger than the time between peaks, so as to produce smooth output with little fluctuations while the signal amplitude remains constant. At the same time, RC should be significantly smaller than the expected speed of change in amplitude, so the peak detector can accurately represent the immediate signal amplitude. In our case, the directional microphones rotate rather slowly, so RC below 1 second is necessary. In all other respects, relatively large values for the resistor and capacitor may be used.

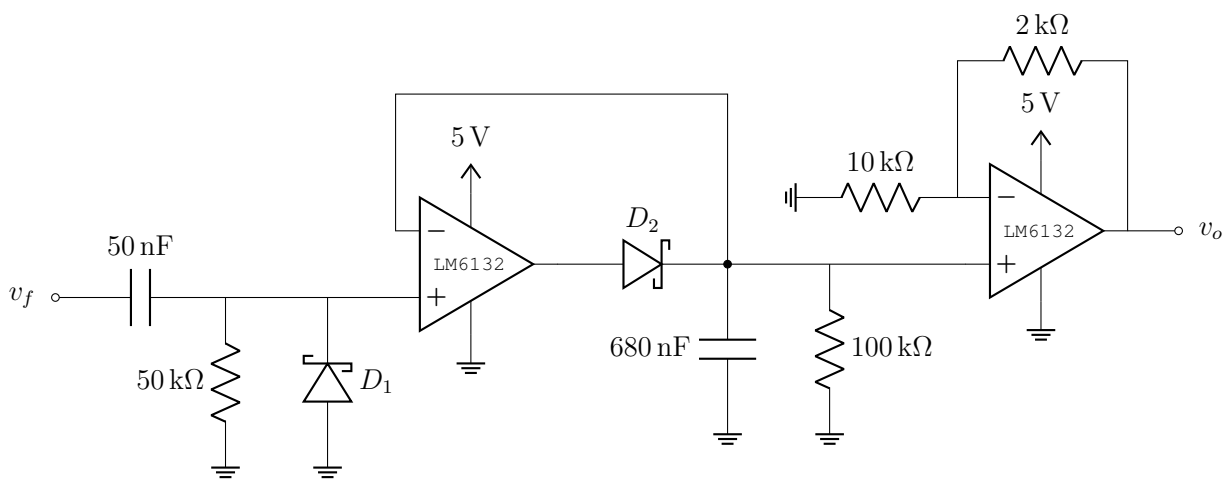

Figure 9: Peak detector as designed

The rest of the components outside of the core three serve to improve the performance and accuracy of the peak detector. Like the amplifier stage, this too has a level shifter right after its input, which is formed by the 50 nF capacitor and the 50 kΩ resistor. However, the level shifter centers the wave back at 0 V to unlock the lower 2.5 V of the 5 V signal range for the ensuing output, maximizing the usable voltage range. The diode D_1 blocks the negative half of the signal after being level shifted to 0 V in order to protect the first op-amp from negative voltages,

which are outside of its supply voltages and can damage the op-amp. The diodes D_1 and D_2 are 1N5817 Schottky diodes instead of regular PN ones due to their lower forward voltages, which is advantageous for each of their roles. The first op-amp in question acts as a buffer, negating the small voltage drop across D_2 which allows the capacitor to charge to the exact voltage of the input peaks, to again maximize the use of the limited 5 V voltage range. The second op-amp has slight amplification to make up for the small losses from the preceding filter and peak detector. At this stage, it is important that the amplifier is non-inverting in order to preserve the information's integrity. The op-amp also acts as a buffer, preventing the peak detector from discharging via the output loading the circuit, to maintain the peak detector's accuracy [16]. At this point, the output from this block is ready to be read by the microcontroller. The behavior of the peak detector when fed a signal simulating a rotating microphone is shown in Figure 10 [17, 18].

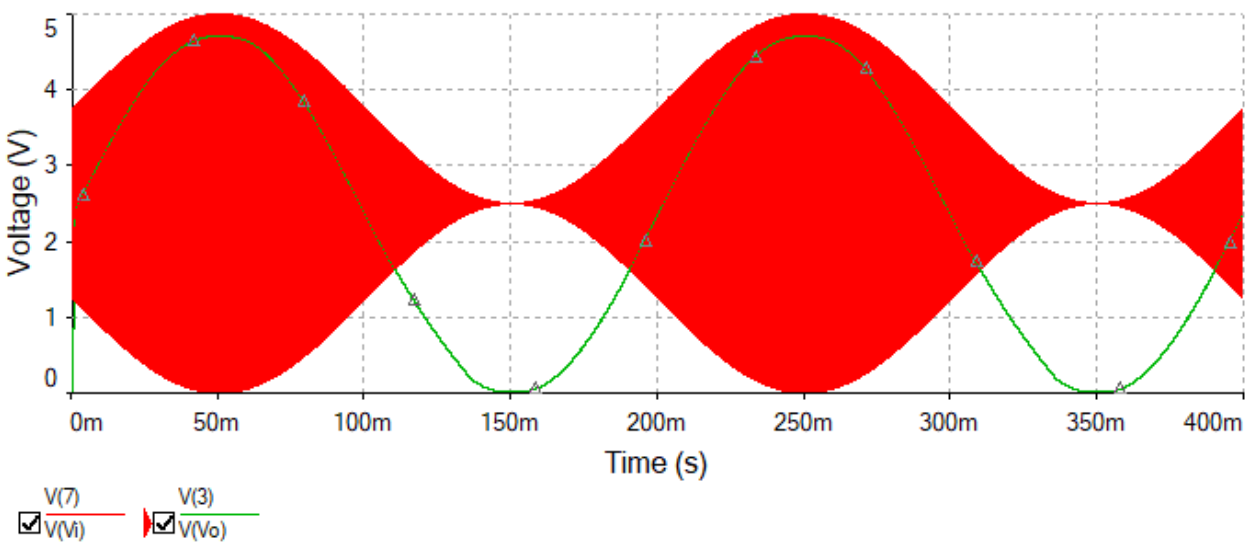

Figure 10: Simulated response of peak detector as designed

3.2 Robot System Design

3.2.1 Robot Body and Dynamics

3.2.1.1 Body Structure

The body of the robot has been designed to contain a platform for the microcontroller, battery, and audio transmission system. The mounted audio transmission system consists of two stepper motors, two mounted microphones, and a printed circuit board (PCB). The PCB has been designed to attach directly to the microcontroller through header pins, so it may be mounted as a single component. In addition to the PCB system, motors and connections must be added for the vehicle's navigation system. Below, Figure 11 shows the connectivity of the robot, and Figures 12 and 13 show orientations of the robot 3D STL model.

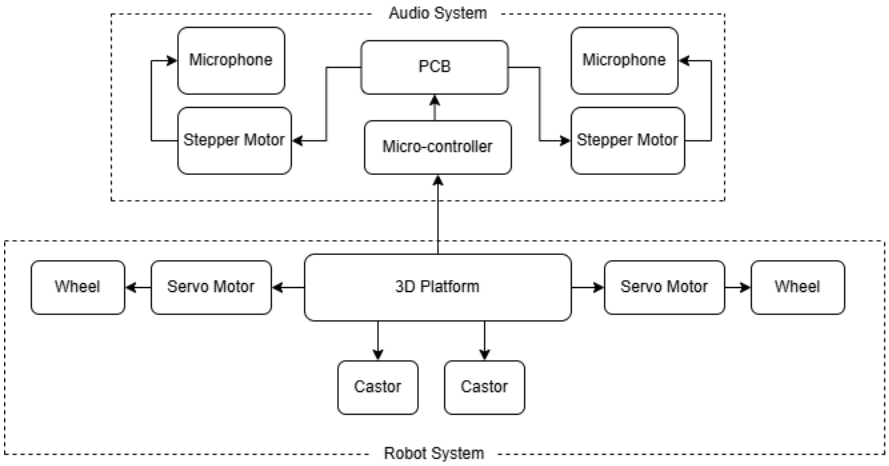

Figure 11: Connectivity of the Robot System

As seen in Figure 12, the bottom of the robot has mounting points on the sides for placement of the continuous servos. Additionally, there are two circular indentations to allow for the mounting of the free rolling castor wheels. The square indentations are placement positions for the driver boards that control the stepper motors.

In Figure 13, the placement position of the PCB and microcontroller block is visible by the rectangular indentation. To host the stepper motors, there are two circular mounting points which the motors fit securely. The wheels and plastic body must be able to support the full weight of the system while still maintaining the ability to navigate accurately.

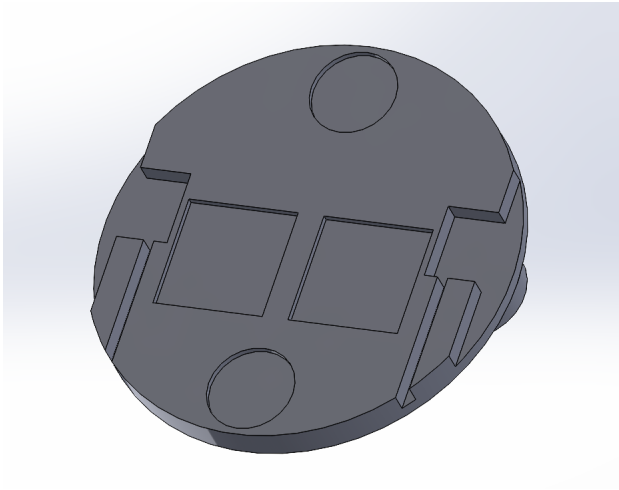

Figure 12: Bottom View of Robot STL Model

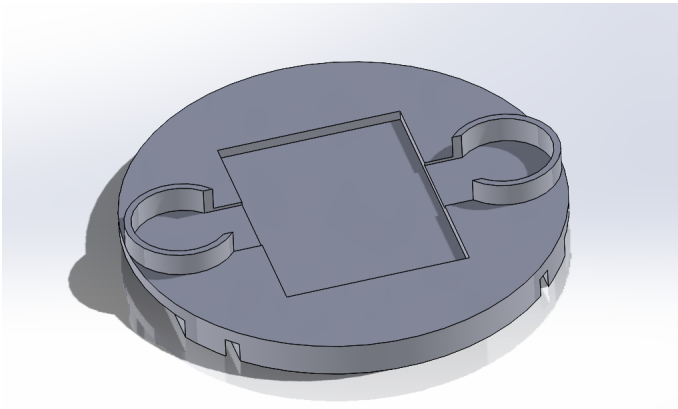

Figure 13: Top View of Robot STL Model

To create an accurate reference for the CAD models in Figures 12 and 13, Table 2 details the dimensions of the robot and the mounted components in terms of maximum height, length, and width from a bounding box around the three-dimensional model.

Table 2: Robot Component Dimensions

	Li-Ion Battery	PCB	Stepper	Servo	MC	Robot Chassis
Length [mm]	50	68	32.85	32.1	68	127
Height [mm]	14	12	32.37	29.95	14	127
Width [mm]	14	50.5	29	12.2	50.5	17

The robot is also responsible for mounting the localization hardware and its corresponding connections. One constraint of robotics for audio localization, is the distance of the microphones in the array. In this robot design, the microphone distance is maximized, while the total robot size is minimized. This results in microphones being on opposite sides, while the PCB, microcontroller, and connections are between them. This idea can be further understood by looking at the 3D model in Figure 13.

3.2.1.2 Robot Navigation

The robot must be able to navigate to the localized sound source on a 2D plane. To do this, the robot uses radial coordinates, where it may rotate an amount of degrees, then travel a necessary distance. The center of the coordinate system will be calculated as the center of the robot; using this, the sound source will always be found relative to the robot's center. This coordinate system may be observed in Figure 14.

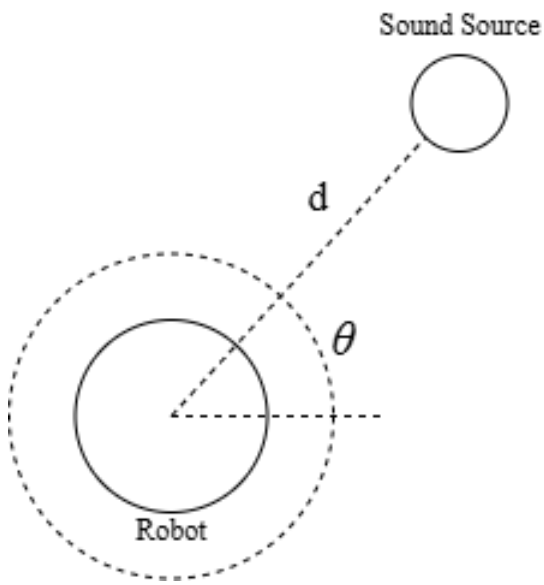

Figure 14: Radial Coordinate System

To realize this, the robot was designed to have two continuous servo motors on its sides, and has two free-rolling castor wheels to provide balance and support. A render of the design may be seen below in Figure 15.

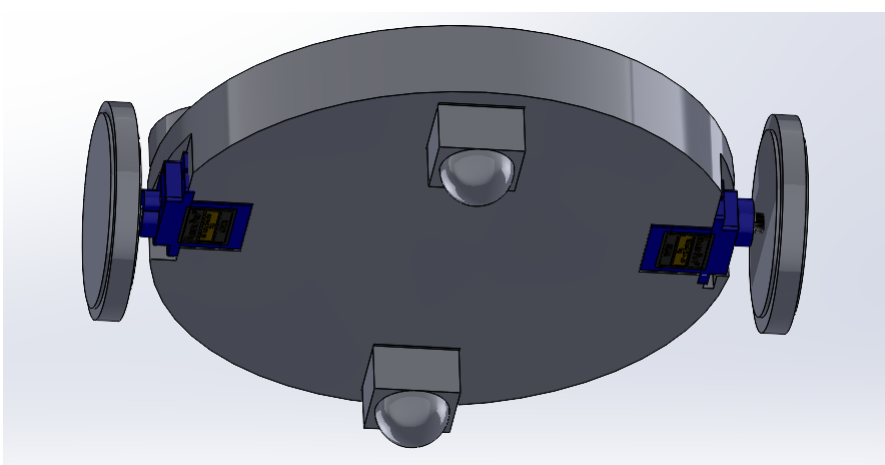

Figure 15: Render of Robot Wheel System

The microcontroller is used to send signals to the robot to turn and travel forward depending on the command. The servo motors only accept signals that dictate the direction and speed of the motor, so rotation and distance must be put into terms of $t_{duration}$, the duration of time that the wheels are turning. For continuous forward movement, both wheels simply move in the same direction at the same speed. To dictate the distance traveled, the time of a single wheel rotation is recorded, and the total distance can be found using the wheel size and time of a single rotation as seen in Equation (7).

$$t_{duration} = d * t_{rotation}/2\pi r_{wheel} \tag{7}$$

For rotational movement, the wheels must move in opposite directions, but the same speed. If the speed of the motors differ at all, then the robot will not rotate around its center. To rotate the robot by θ degrees, then each wheel must equally rotate θ degrees as well, but in opposing directions. Using the received measurements, the rotation may be put in terms of the time of wheel rotation using the angle of rotation and wheel size as seen in Equation (8).

$$t_{duration} = t_{rotation} * \frac{r_{robot}}{r_{wheel}} * \frac{\theta}{360^o} \tag{8}$$

With the proposed design, the microcontroller should send a command containing data for θ and d, then the navigation system should move the robot to the corresponding location. Figure 16 details the process for taking a navigation command and performing accurate movement.

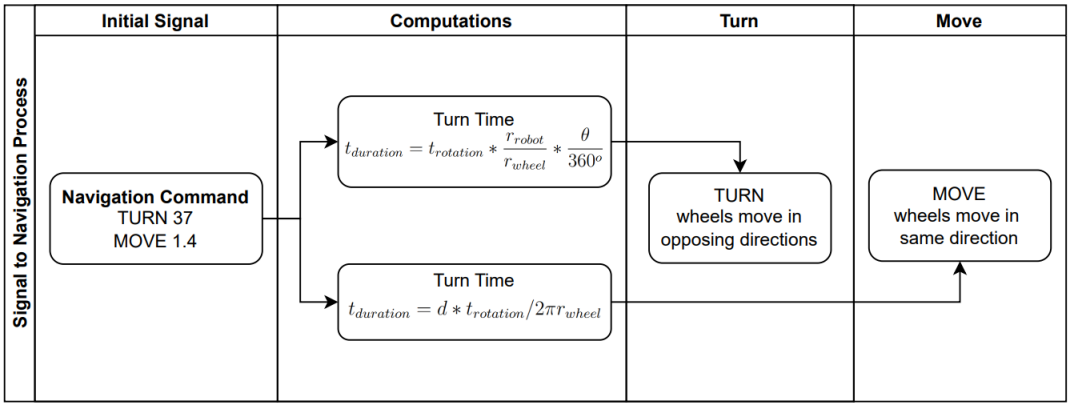

Figure 16: Process from Signal to Navigation

For the communication interfaces, the servo motors have a single data wire that connects to the microcontroller through a pin on the designed PCB. The microcontroller is then programmed to send a PWM signal to transmit the command, θ, and d data values through the aforementioned connections.

3.2.2 Microphone Enclosures

In addition to the stepper motors and microphones, the design allows for microphone enclosures to impact the directionality and accuracy of the localization algorithm. In Figure 17, the mounting structure for the microphone to stepper motor may be seen.

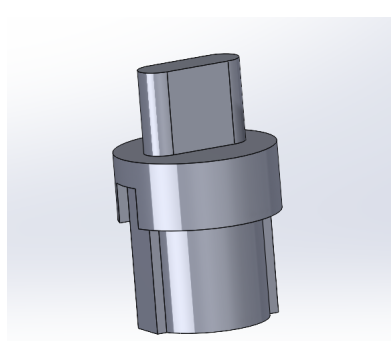

Figure 17: Microphone Enclosure Mount

The microphone enclosures may be connected using the peg at the top of the structure. For the motor connection, a cavity at the bottom was designed. To illustrate, Figure 18 shows a rendering of the enclosure mount attached to the stepper motor.

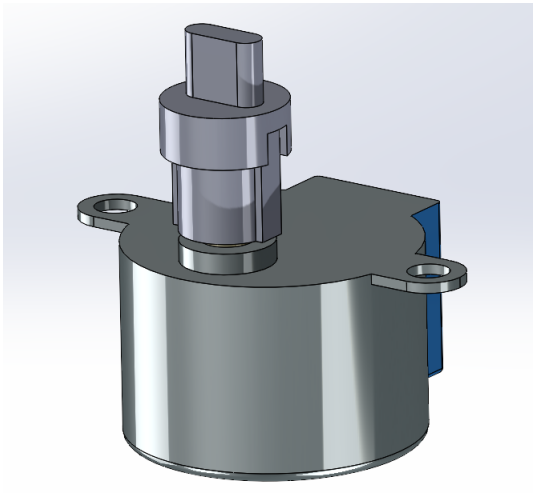

Figure 18: Mounted Enclosure Structure

The purpose of the microphone enclosures is to increase the directionality to a point where received signal is "high" when the microphone is in perfect alignment with the sound source, and is "low" otherwise. Considering this, the enclosures were designed to maximize directionality and minimize the received signal when misaligned.

For the first enclosure, a barrel-like structure was designed. The barrel model is intended to physically block sound coming from directions when the sound source and microphone are not directly facing. With a barrel thickness of 2 millimeters, misaligned sound will be attenuated to a significant degree without causing mechanical complexity. For a barrel length extending at least the diameter of the microphone, the directionality is maximized. Figure 19 contains the 3D model used to print the structure.

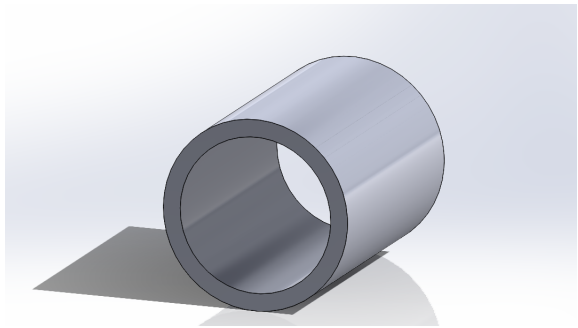

Figure 19: Barrel Enclosure 3D Model

The resulting directionality and localization accuracy of this enclosure may be observed in the experiment section of the paper. Figure 20 contains a picture of the enclosure mounted to a microphone and stepper motor.

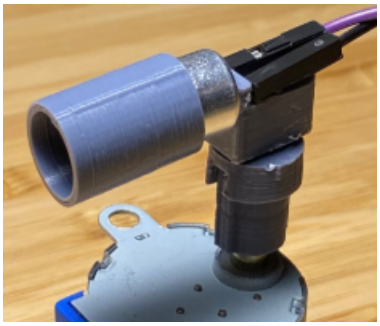

Figure 20: Picture of Mounted Barrel Enclosure

For a more advanced enclosure, the parabolic model was designed. This model uses a parabolic dish to focus the received sound at a focal point directly into the microphone. Since our design requires horizontal directionality, the vertical directionality would act as a constraint. Taking this into account, the top and bottom of the parabolic enclosure were clipped to reduce the negative impact of this effect. Additionally, the parabolic structure will only work properly if the

unenclosed microphone accepts audio signals from a very low directionality. Figure 21 contains a diagram showing the functionality of a parabolic structure to focus a signal to a focal point.

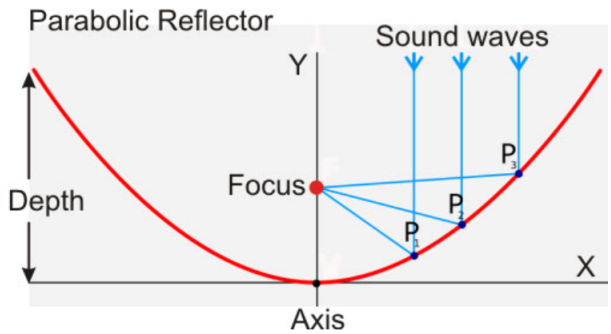

Figure 21: Parabolic Focusing

For this specific enclosure, the dish depth is 12.66 mm and the dish diameter is 46.02 mm. Using these dimensions, the distance of the focal point from the dish center may be calculated using Equation (9), where f is the diameter of the dish and a is the depth of the dish [19].

$$h = \frac{a^2}{16f} \tag{9}$$

Using this calculation, the focal point was found to be 10.46 mm from the dish [20]. Using these design parameters, several parts were designed to achieve the parabolic dish rendered in Figure 22. Additionally, the printed and mounted model may be seen in Figure 23.

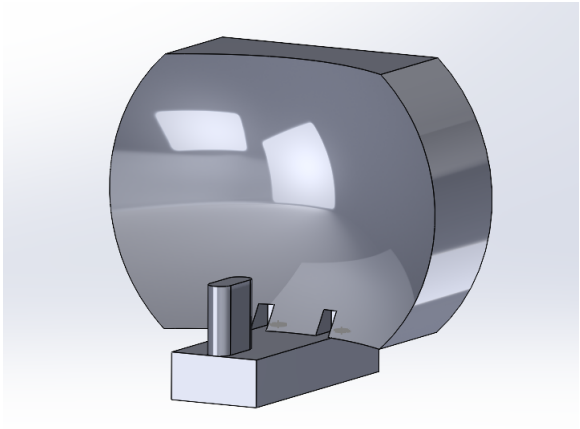

Figure 22: Parabolic 3D Model

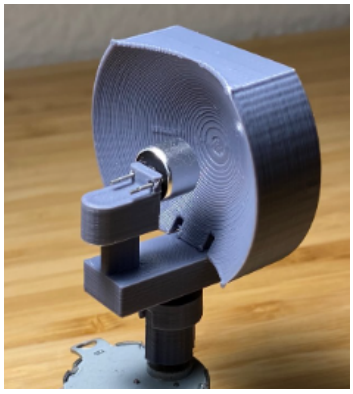

Figure 23: Mounted Parabolic Enclosure

3.2.3 Power Distribution

Since the robot must be entirely mobile, a battery will be used to power the full system. In our design, a 7.4 V lithium-ion battery was chosen because of the battery lifetime and the drop out voltage required by the circuit's voltage regulator. This battery powers the microcontroller, two stepper motors for microphone rotation, and two servos for movement. The battery is used to power each object in Table 3 which contains the voltage requirements and the average current

draw of each respective component.

Table 3: Voltage and Current of Components

	Lithium Ion Battery	PCB	Stepper	Servo	Microcontroller
Input Voltage [V]	-	7.4	7.4	5	5
Current [A]	-	0.044	0.549	0.128	0.028
Output Voltage [V]	7.4	5	-	-	-

In the following graphic, Figure 24 demonstrates the power distribution layout and includes the voltage requirements and current draw at each load.

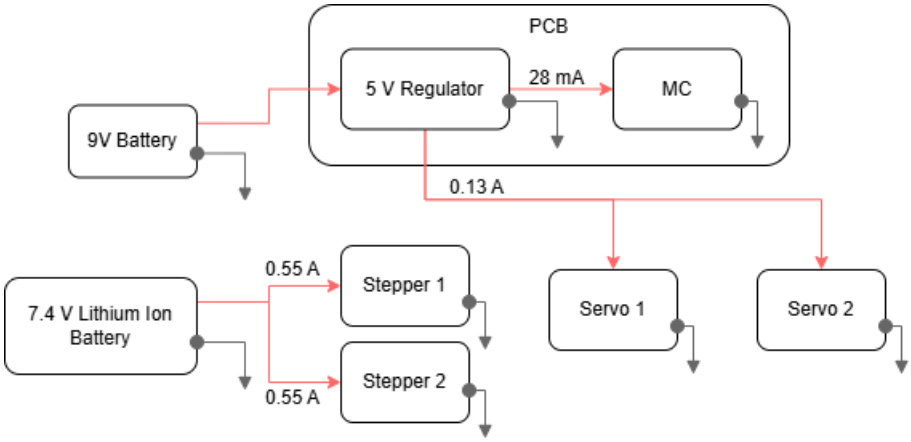

Figure 24: Power Distribution

For the servo motor, the Fitech FS90R was selected [21]. These continuous servo motors have controls that depend on the supplied voltage. Since these motors are used for the wheels, if we are to have consistent and equal movement between the wheels, the supplied voltage must be equal and regulated. To satisfy this requirement, the audio transmission PCB will have an integrated voltage regulator to ensure a constant 5 V supply to the servos.

For the stepper motors, the battery may be directly connected to their respective driver boards. The 28BYJ-48 stepper motor is being used due to its voltage requirements, price, and connectivity with the ULN2003 driver board. The maximum current output of the battery will not be exceeded [22, 23]. When the wheels are active, the current draw will be 260 mA; when the steppers are in operation, the current draw will be 1.1 A.

3.3 Algorithm Design

The design of the localization algorithm is guided by the need to reduce computational complexity while maintaining high accuracy in sound source localization. The algorithm is specifically tailored for robotic applications, where processing power and physical space are often limited. Key design considerations include the use of rotating microphones, the implementation of trigonometric calculations for distance estimation, and the incorporation of peak detection to ensure precise angle measurements. To accomplish these requirements, the Arduino Uno R3 was selected to run the algorithm software and communicate with peripherals [24].

The use of rotating microphones is a central feature of the algorithm. By rotating the microphones, the system can achieve a higher degree of directionality, allowing for more accurate angle of arrival measurements. The rotation is controlled by stepper motors, which are programmed to sweep through a predefined range of angles while recording the amplitude of the sound signal. The angle at which the maximum amplitude is detected corresponds to the direction of the sound source. This approach eliminates the need for a large array of microphones, reducing both the hardware complexity and the computational load.

Due to the heavy reliance on accurate angle measurements, simulations were performed early in the design process to characterize the directionality behavior. Figure 25 shows the importance of ensuring directionality with a microphone enclosure as if the robot is able to detect sound sources from behind it would give false peaks as shown in difference of the center of the two orange plots in Figure 25.

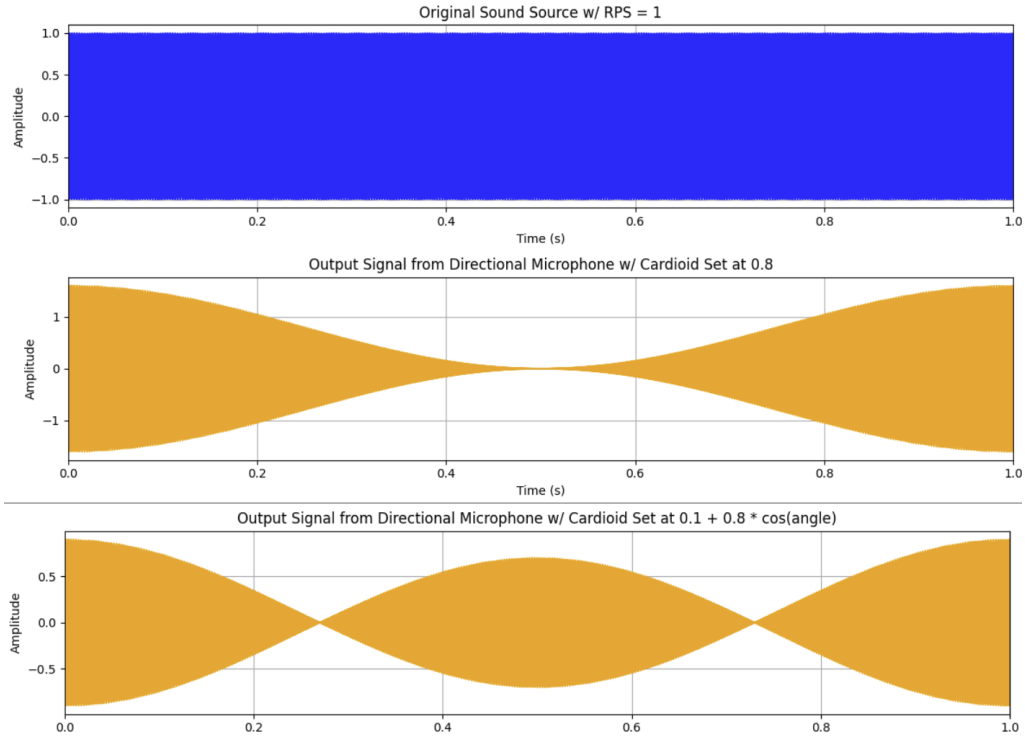

Figure 25: Directional Cardioid Simulations

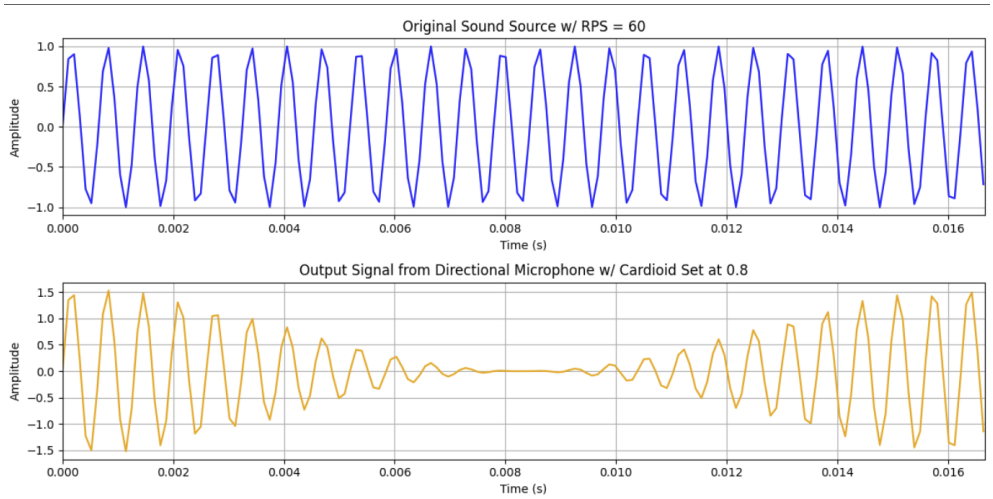

Figure 26: 60 RPS Simulation

Simulations also indicated significance in the tradeoff between speed and accuracy of localization. Figures 25 and 26 show the significant difference in the quality of gathered data between a rotation speed of 1 RPS and 60 RPS. These simulations gathered valuable insight into constraint considerations and provided a proof of concept for the sweep method before hardware development began.

To ensure precise angle measurements, the algorithm incorporates a peak detection mechanism. The peak detector is implemented in the analog frontend of the system, where it converts the carrier wave of the sound signal into a simple voltage representing the signal's amplitude. This voltage is then digitized and fed into the microcontroller, where the algorithm identifies the peak amplitude and the corresponding angle of arrival. The use of peak detection allows for a reduced sampling frequency and simpler reception of amplitude data.

One of the primary goals of the algorithm design is to reduce computational complexity. Unlike traditional TDoA methods, which require complex cross-correlation calculations and large arrays of microphone data, the proposed algorithm relies on simple trigonometric calculations and peak detection. This technique makes the algorithm well-suited for implementation on low-power microcontrollers, which are commonly used in robotic systems. Additionally, the algorithm is designed to operate in real-time, allowing the robot to quickly localize and navigate toward the sound source.

The localization algorithm system's implementation has three hierarchical modes, each designed to address specific operational constraints. The following describes the proposed algorithms and details design decisions and reasoning.

3.3.1 Naive Algorithm

The simplest approach, termed the Naive Peak Detection Algorithm, functions by taking the global maximum value from both microphones after a single complete 360° sweep. While this is the most computationally efficient, it is vulnerable to saturation at close range and slow operation for significant data collection. If there are many angles showing as the maximum value, using the Naive Peak Detection Algorithm will fail in giving a reliable reading. Therefore, another mode is

created as an additional layer for when the maximum possible value is detected at more than one step.

3.3.2 Averaged Maxima Algorithm

The Averaged Maxima Algorithm introduces robustness by calculating the central value of amplitude readings containing the maximum possible value. This smoothing mechanism filters spurious peaks caused by impulsive noise, keeps computational load light, and allows for finding direction through a saturated peak signal around the true location.

3.3.3 Slicing Algorithm

The Slicing Algorithm cuts the rotation into N evenly spaced points along the motors circumference. The microphone then rotates to each point and collects a large number of measurements which are averaged to diminish noise. It may then be estimated that the angle of the sound source lies between the two points with the largest (point 1) and second largest (point 2) measured amplitude. Now, the window from point 1 to point 2 may be cut into N slices and the process can repeat. This process will continue until points 1 and 2 are the same position, or until the measured amplitudes of 1 and 2 are equal. If the points are the same position, then this angle will be predicted. If the points are not the same position, but have the same amplitude, then the angle will be estimated as their average. This method reduces the number of angles measured and exploits the extra time to collect a higher volume of measurements. Noise is averaged out by collecting a high volume of measurements for each angle of interest to improve total accuracy.

3.3.4 Distance Estimation

After the angle is determined from each microphone, triangulation may be used for an estimation of the sound source distance. The triangulation accuracy is highly dependent on the accuracy of the angle approximation of the aforementioned algorithms. The last step of each of the three algorithm modes is to take the calculated peak angles to send a radial coordinate as depicted in Figure 14 through the microcontroller for the motors to have accurate pathing data to move the correct distance and direction to the ultrasonic sound source.

3.4 Integration

The algorithm subsystem's integration with other components was carefully optimized to minimize latency and resource contention. Peak-detected amplitude signals from the analog frontend were read via the Arduino's ADC pins (A0, A1) at around 9.6 kHz. Motor commands were issued through pins 8 and 9. Navigation coordinates in the terms of distance and angle were transmitted to the wheel servos (pins 5, 6) and scaled to account for the robot's dimensions. Integration testing revealed a critical insight: noise from the stepper motor was introduced to the analog microphone signal. Isolating the stepper motor's power supply mitigated this effect.

The fully developed PCB may be seen in Figures 27 and 28. After full development and integration, a compact and effective robot was designed and may be seen in Figures 29, 30, and 31. As seen in the images, the robot model mounts the wheels, stepper motors, servo motors, driver boards, PCB, microcontroller, and microphone enclosure. In the given image, the parabolic enclosures are mounted.

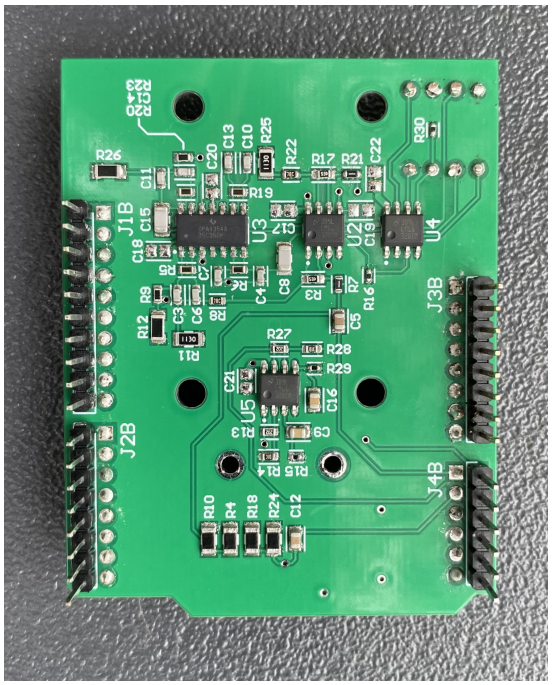

Figure 27: Picture of the Back of the PCB

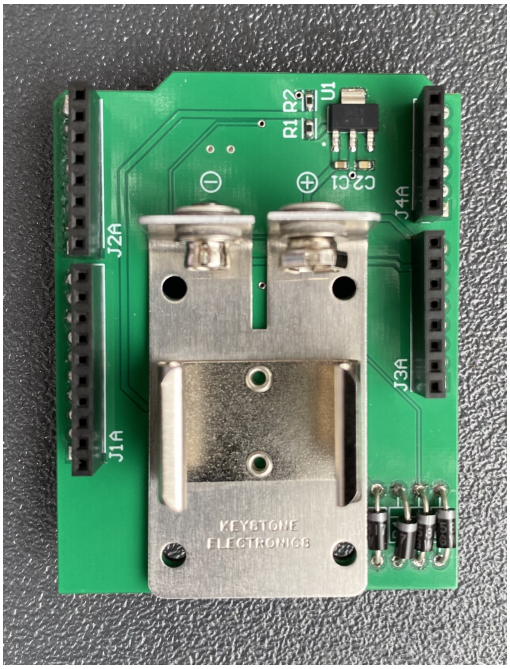

Figure 28: Picture of the Front of the PCB

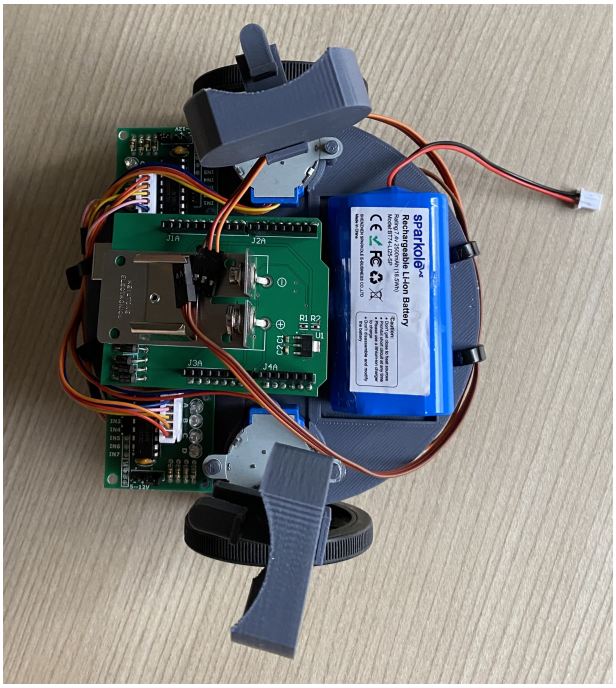

Figure 29: Top View of Robot

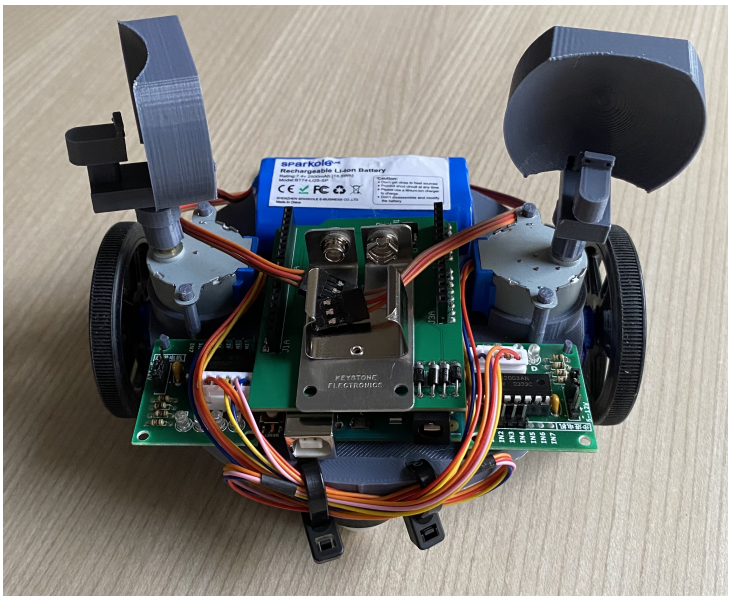

Figure 30: Front View of Robot

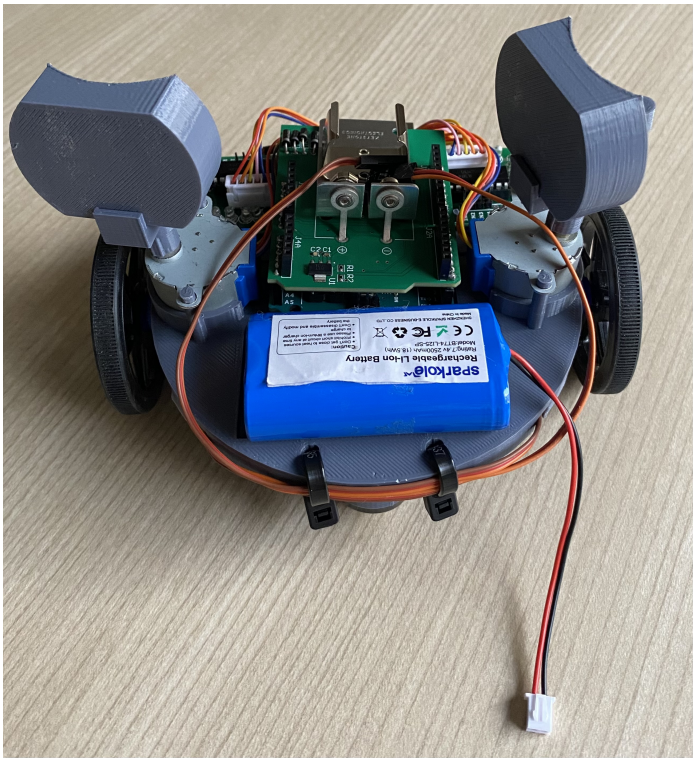

Figure 31: Back View of Robot

4. VALIDATION

4.1 Analog Frontend

To validate the functionality of the designed PCB, the amplifier to peak detection system was checked against the expected behavior. Additionally, it was verified that the board met the desired specifications as seen in Table 4.

Table 4: Analog Frontend Validation

Property	Specification	Implementation Result
Signal-to-noise ratio	$\geq 40\,\mathrm{dB}$	$51.447\,\mathrm{dB}$
Maximum output voltage swing	$\geq 3.5\,\mathrm{V}$	$4.972\,\mathrm{V}$

To validate the amplification and peak detection circuitry, a sinusoidal input was applied at various peak voltages and the output was compared with the expected, ideal output. This information may be seen in Table 5.

Table 5: Peak Detector and Amplifier Validation

Input Peak [V]	**Ideal Output [V]**	**Measured Output [V]**	**Error %**
0	0	0	0
0.01	0.35	0.334	4.67
0.03	1.05	1.034	1.52
0.05	1.75	1.735	0.857
0.07	2.45	2.434	0.653
0.09	3.15	3.133	0.54
0.11	3.85	3.831	0.49
0.13	4.55	4.532	0.396
0.143	5	4.973	0.54

4.2 Robot Validation

4.2.1 Distance Validation

To validate the distance, various distances were sent to the microcontroller, then the corresponding distance that the robot traveled was measured. Table 6 shows the obtained measurements and error of the distance traveled.

Table 6: Robot Distance Validation

Expected Distance (m)	**Measured Distance (m)**	**Error**	**Error %**
0.25	0.25	0	0
0.5	0.493	0.01	1.4
0.75	0.742	0.01	1.07
1	0.99	0.01	1.0
1.5	1.479	0.02	1.4
2	1.948	0.05	2.6
3	2.932	0.07	2.27
4	3.911	0.09	2.225
5	4.88	0.12	2.4

4.2.2 Turning Validation

To validate the angle being turned to, the robot was pointed coincident to a line of tape. Commands were sent to the microcontroller to rotate the robot by a number of degrees, then the true angle was measured using a digital protractor from the tape, to the new position. Table 7 shows the measurements taken and the respective error calculated.

Table 7: Robot Turning Validation

Expected Angle	Measured Angle	Error	Error %
30	29.88	0.12	0.4
60	59.94	0.06	0.1
90	90.03	-0.03	0.0333
120	120.14	-0.14	0.117
150	150.23	-0.23	0.153
180	180.37	-0.37	0.206

4.3 Microcontroller Validation

After validation of the analog circuitry and robot system, the proper reception of digital signals must be verified. To test this, an LCD screen was used and attached to the developed PCB as seen in Figure 32.

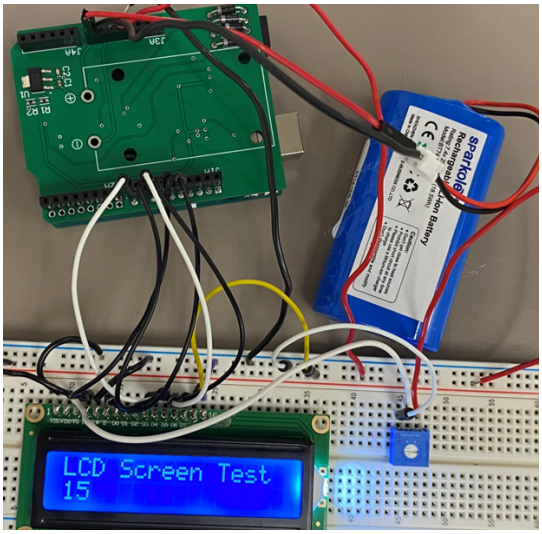

Figure 32: LCD Setup for Digital Validation

By adding a microphone to the setup in Figure 32, it is used to validate the received digital signal. By mimicking the validation process for the analog circuit, the results in Table 8 were obtained for the digital signal.

Table 8: Digital Signal Validation

Input Peak [V]	**Expected Output**	**Measured Output**
0.000	0	0
0.909	186	183
1.818	372	368
2.727	558	555
3.636	744	740
4.545	930	927
5.000	1023	1017

5. EXPERIMENTATION

5.1 Microphone Enclosure Experiment

For the microphone enclosure experiment, two significant data metrics were recorded for each enclosure. These metrics are designed to describe two behaviors: the directionality and distance. This data allows for the characterization of each enclosure to be used for different distances and localization algorithms.

5.1.1 Unenclosed

5.1.1.1 Audio Sweep

This portion of the experiment shows the native directionality of the microphone by plotting the measured amplitude at each angle. Since the stepper motor is limited to 2052 steps and the quantizer is limited to 1024 values, the plot is not continuous in the X or Y axis. The plot is constructed by performing a full 360 degree audio sweep 1 meter from the sound source, with a full rotation taking 30 seconds. The functional speed for measurements is slower than the system will be in operation, so that we can obtain baseline measurements with low error. Additionally, the sound source must be facing the robot chassis due to the high directionality of the 40 kHz sound source. If the sound source were to be misaligned, the angle of arrival can not be reliably predicted unless the sound source is minimally directional. In Figure 33, the unenclosed microphone behavior may be seen on the X-Y plane, while Figure 34 is a radial representation.

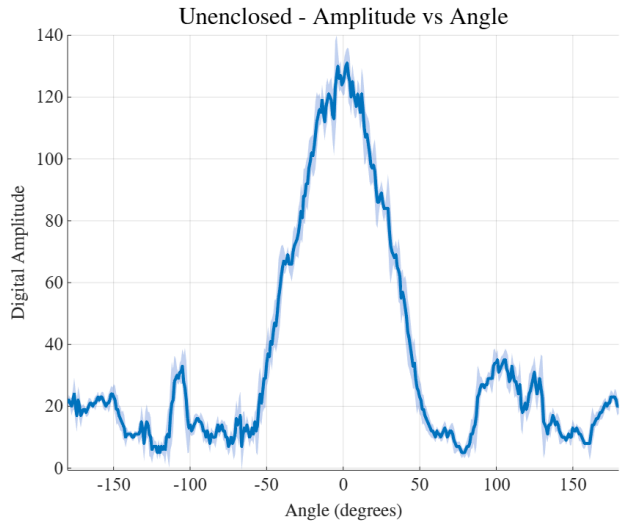

Figure 33: Directionality of Unenclosed Microphone (XY-coords)

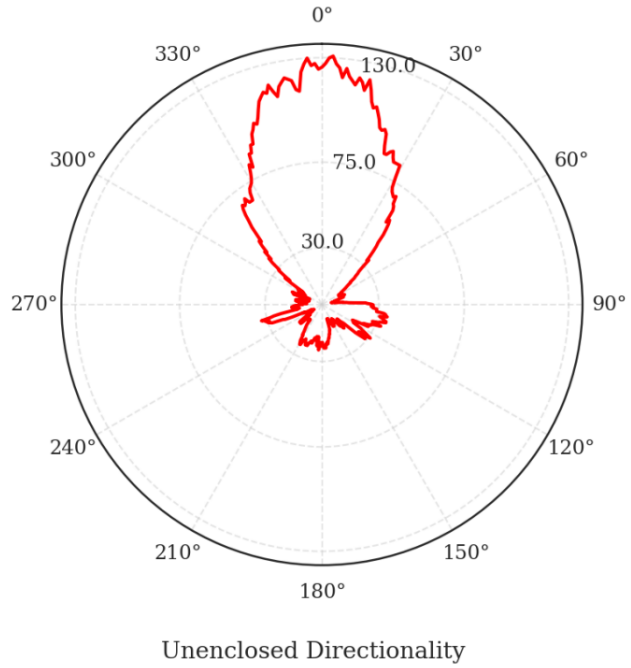

Figure 34: Directionality of Unenclosed Microphone (radial-coords)

5.1.1.2 Distance Behavior

To characterize the distance capabilities, the amplitude at direct alignment and direct misalignment were measured until exponential attenuation was observed. Figure 35 displays this information and shows the effective distance for which the robot is functional.

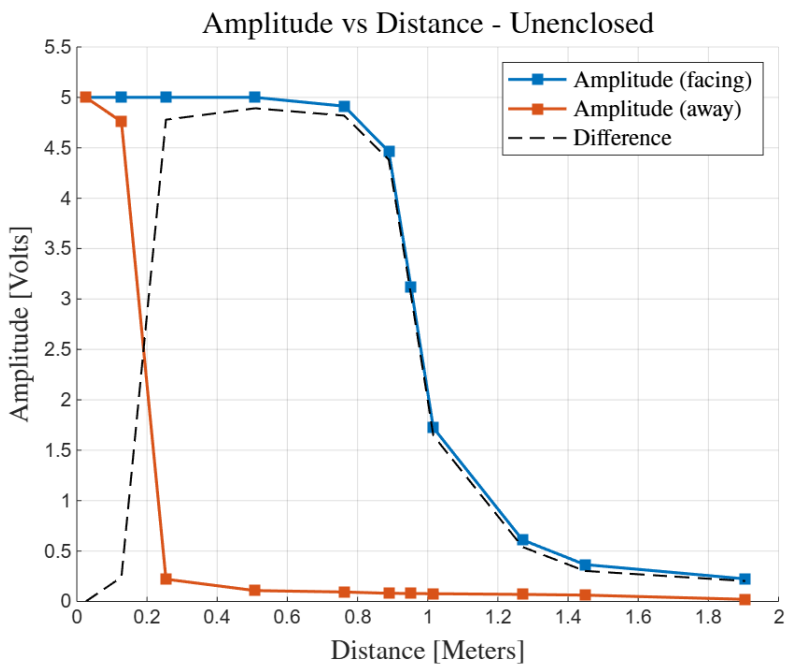

Figure 35: Distance vs Amplitude - Unenclosed

5.1.2 Barrel Enclosure

5.1.2.1 Audio Sweep

Equipped with the barrel enclosure in Figure 20, the directionality was observed to be higher. Additionally, when the microphone is not facing the sound source, the enclosure seems to successfully block out unwanted signal. Using the same test setup as the unenclosed microphone, the directionality results in Figures 36 and 37 were obtained.

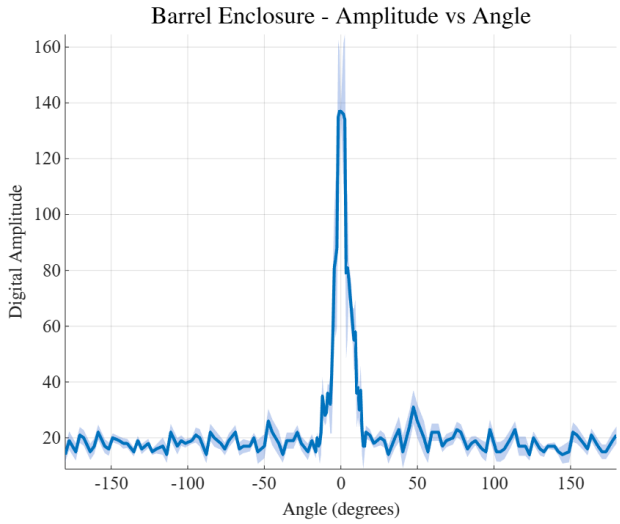

Figure 36: Directionality of Barrel Enclosure (XY-coords)

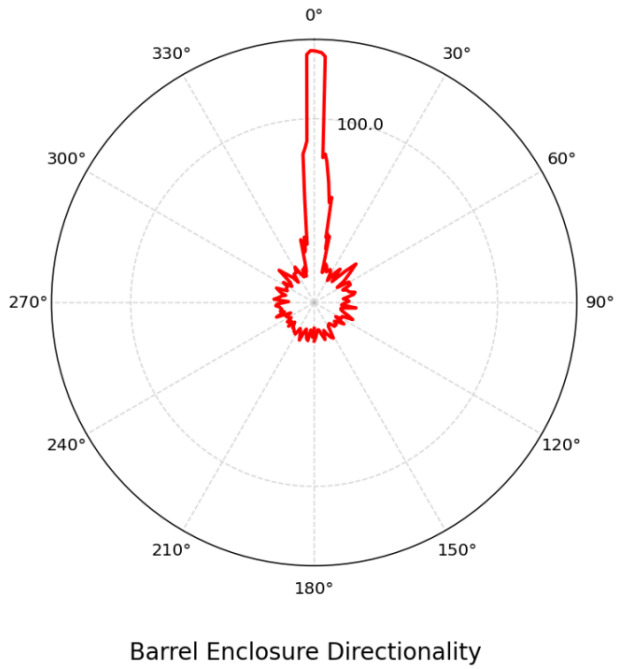

Figure 37: Directionality of Barrel Enclosure (radial-coords)

5.1.2.2 Distance Behavior

As seen in the distance plot, the barrel enclosure does not implement any improvement to the maximum or minimum distances. Figure 38 depicts the obtained results and shows no significant change from the unenclosed microphone.

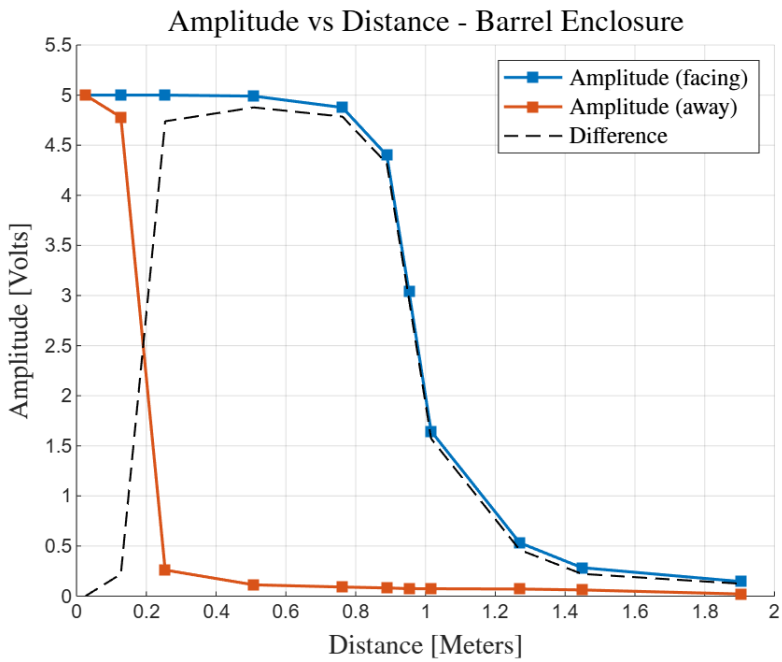

Figure 38: Distance vs Amplitude - Barrel Enclosure

5.1.3 Parabolic Enclosure

5.1.3.1 Audio Sweep

From the parabolic enclosure, a heightened directionality may be observed along with a strong rejection of misaligned audio signals. Without any changes to the experiment, when the parabolic enclosure is used, the measurements in Figures 39 and 40 were obtained.

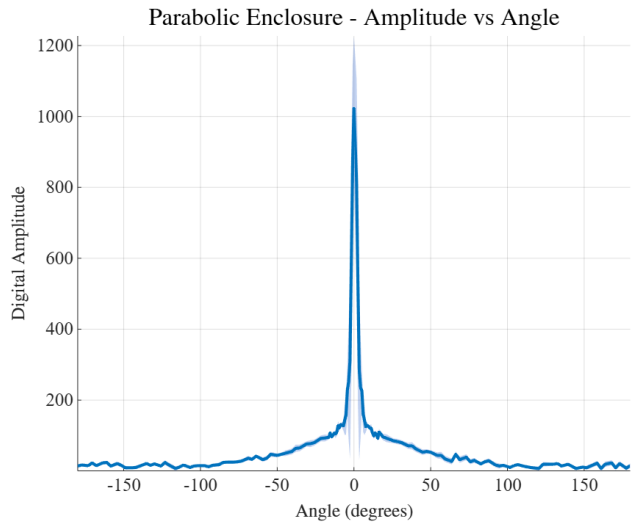

Figure 39: Directionality of Parabolic Enclosure (XY-coords)

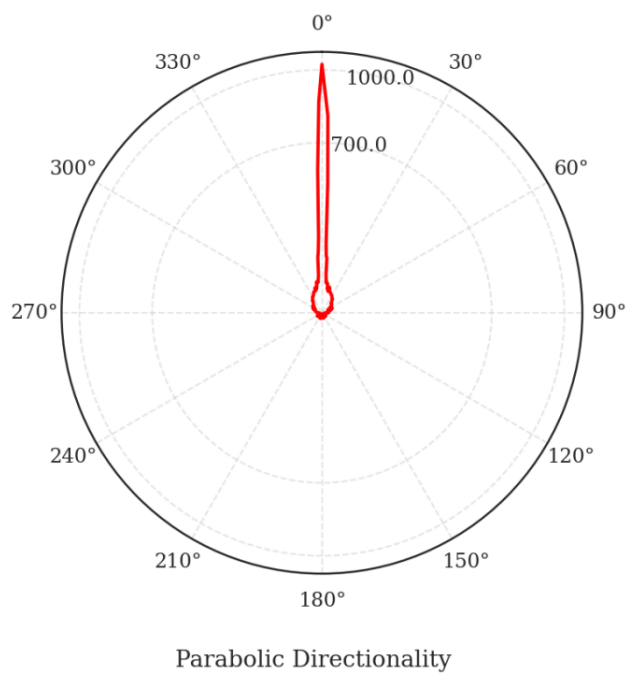

Figure 40: Directionality of Parabolic Enclosure (radial-coords)

5.1.3.2 Distance Behavior

From the distance experimentation, it may be seen that the maximum distance is greatly increased by the parabolic dish. The results may be seen in Figure 41 which show that the effective range of the microphone is nearly doubled when compared to the barrel enclosure and unenclosed microphone.

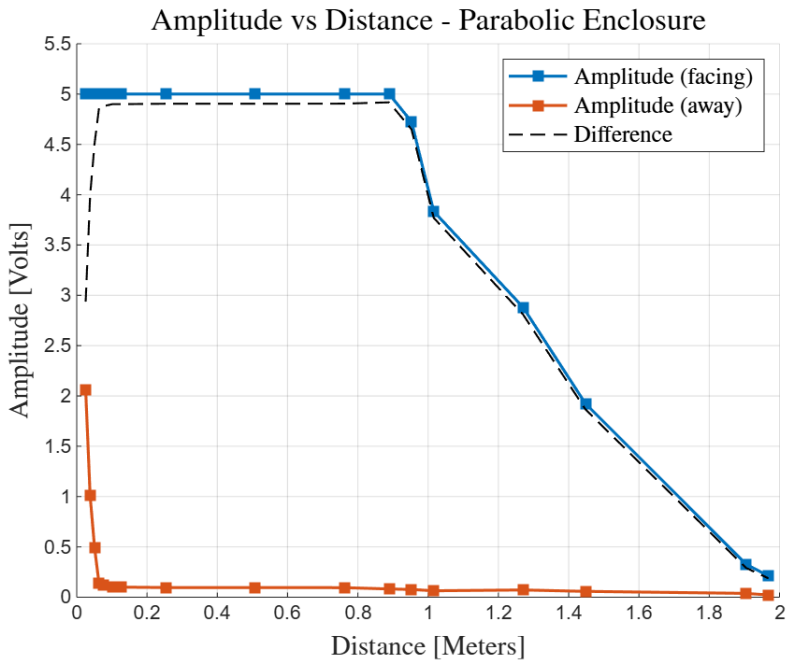

Figure 41: Distance vs Amplitude - Parabolic Enclosure

5.2 Algorithm Experiment

This paper introduces three algorithms for localization: the naive approach, averaging algorithm, and slicing algorithm. In the algorithm experiment, the accuracy of each localization algorithm is analyzed relative to each microphone enclosure. Additionally, the trade-off of time and accuracy is analyzed along with the accuracy over distances in terms of signal to noise ratio (SNR) [25]. For the test functionality, the high frequency sound source is facing directly towards

the robot, but the robot is still unaware of the angle of arrival until after localization. This is important due to errors and ineffective localization that can occur from the directional sound source not reaching the microphone as intended. This section details the overall effectiveness of the proposed method in this paper and details analysis of parameters to improve metrics.

First, to test the accuracy of each configuration at different SNRs, the robot navigation was disabled and set at a distance from the sound source to achieve the desired SNR. Then the algorithm was ran for 10 iterations and the average accuracy is recorded. This experiment allowed for an analysis of each algorithm for different noise ratios or distances from the sound source.

5.2.1 Naive Algorithm

As discussed in the design section of this paper, the naive algorithm performs a sweep and records the angle where the greatest amplitude occurs. The following section details the accuracy analysis for this methodology. In Figure 42, the accuracy of the algorithm may be observed for each microphone enclosure at various distances. As for distance estimation, if using triangulation methods, the accuracy of distance estimation will be ineffective using the naive algorithm.

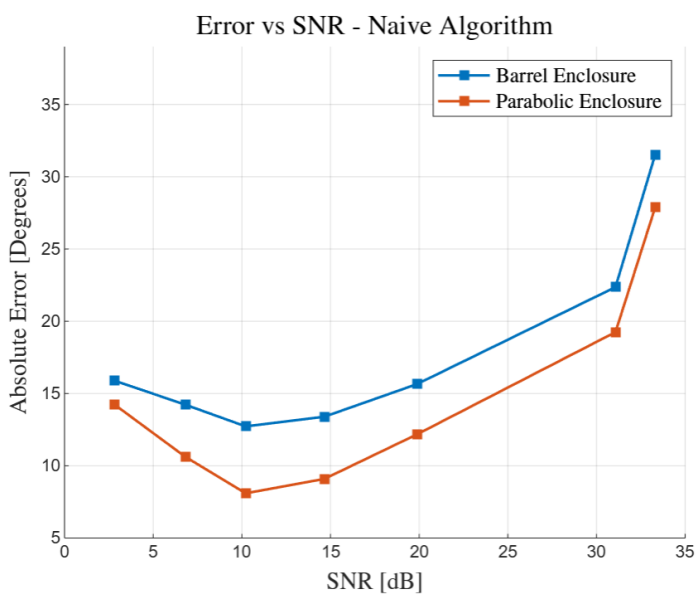

Figure 42: Error vs Distance - Naive Algorithm

5.2.2 *Averaging Algorithm*

For the averaging algorithm, the accuracy for each enclosures at varying SNRs may be seen in Figure 43. For distance estimation when using the averaging algorithm, the barrel enclosure is ineffective for distances 0.195 meters or greater. With this algorithm, the parabolic reflector may triangulate up to 0.313 meters away.

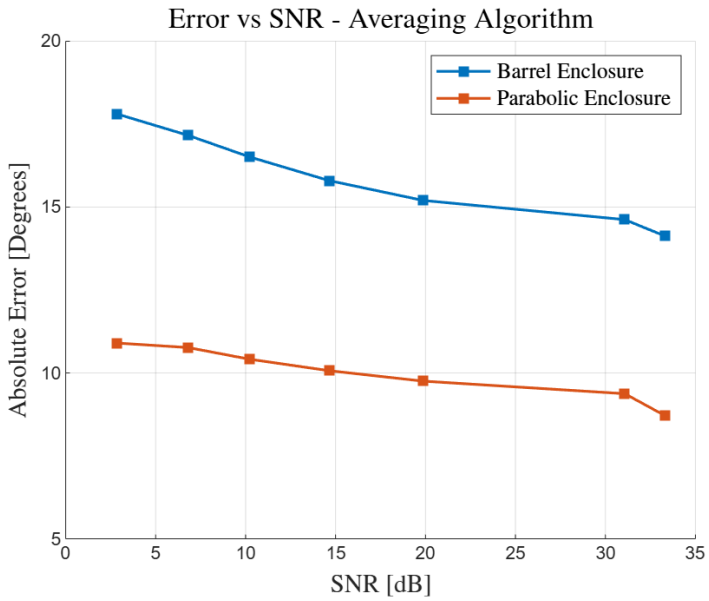

Figure 43: Error vs Distance - Averaging Algorithm

5.2.3 *Slicing Algorithm*

Since the slicing algorithm focuses on fewer points, it is able to collect more measurements of a single point to average out insignificant errors. Figure 44 shows the error of the slicing algorithm equipped with the barrel enclosure and parabolic enclosure. The slicing algorithm yields a maximum distance estimation of 0.365 meters using the barrel enclosure and 0.519 meters using the parabolic enclosure. Figure 45 shows the tradeoff effects of localization speed vs accuracy where it can be seen that past an operation time of 12 seconds, the absolute error does not diminish significantly.

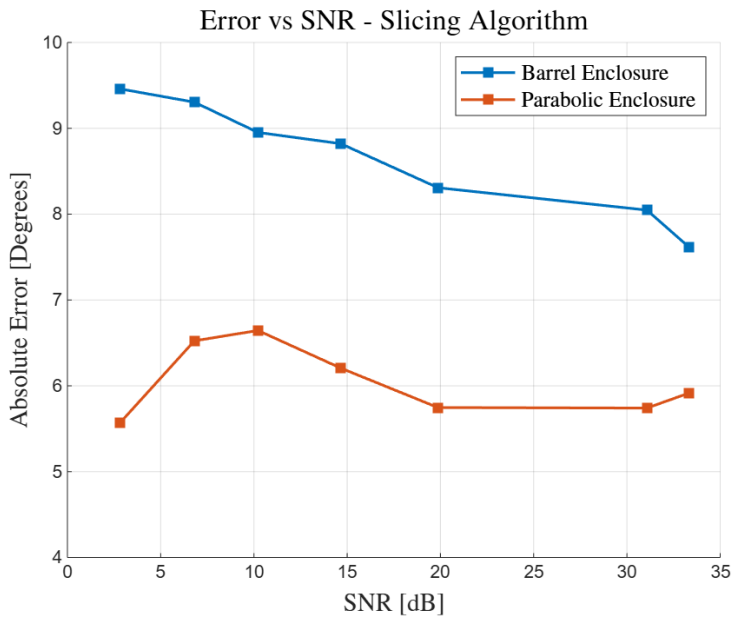

Figure 44: Error vs Distance - Slicing Algorithm

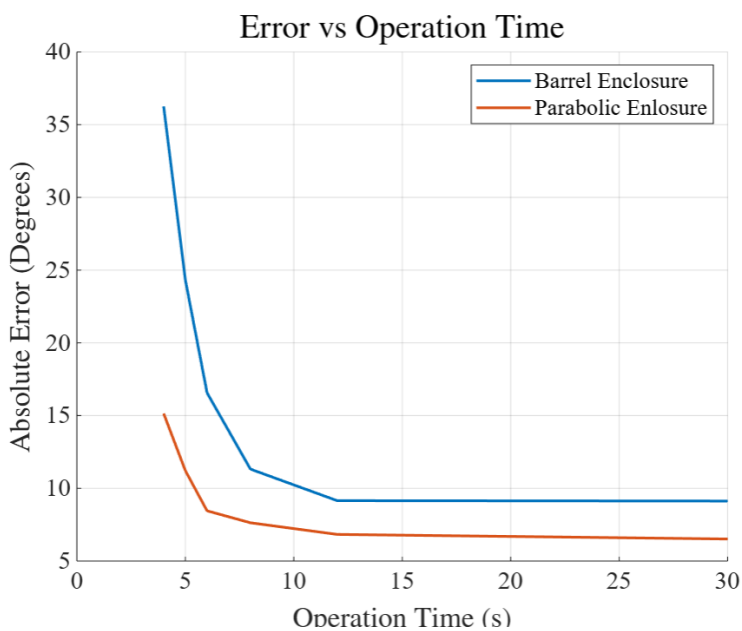

Figure 45: Time to Accuracy Tradeoff

6. RESULTS

6.1 Angle Approximation

After compiling the data gathered during the experiments, Figure 46 was created to display the overall accuracy of using rotation for audio localization. The results indicate that a highly directional microphone will increase the accuracy of localization in most circumstances. The parabolic enclosure achieved higher accuracy when compared to the barrel enclosure which has a lower directionality. Regarding the algorithms, it was expected that a more involved algorithm will increase the accuracy, but the design goal was to minimize computation. The slicing algorithm provided a low computation algorithm that performs better than the other approaches. For the highest accuracy combination—the parabolic enclosure and slicing algorithm—the accuracy is comparable to advanced localization when tested under the same SNR [2]. Additionally, the achievement of reasonable accuracy required systematic errors from the sound source directionality to be removed through alignment of the source and the robot. Overall, angle of arrival may be approximately determined through rotating microphone systems, but more exploration is needed for minimally directional speakers, and for other frequency ranges.

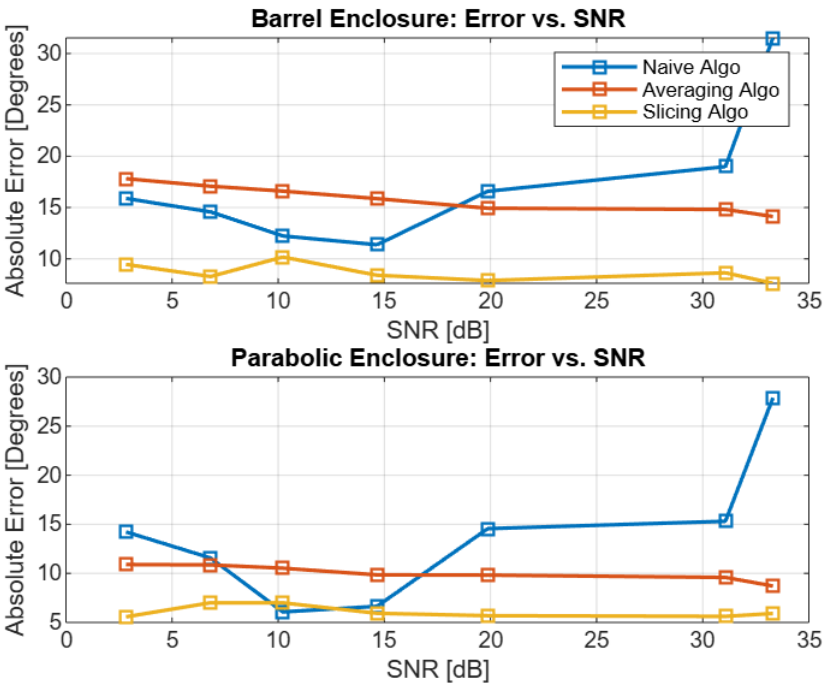

Figure 46: Error Results for Various Experiments

6.2 Distance Estimation

For each combination of microphone enclosure and localization algorithm, triangulation seems to be ineffective for two rotating microphones. Error in the angle estimation of one angle will create a greater error in the distance estimation which makes the method impractical for this implementation. For a more effective implementation, audio intensity methods may be used to estimate the distance if calibrated for the application. Usage of gain control circuitry may also increase the effective distance and estimation accuracy. Additionally, since the angle of arrival for two points are approximated, more complex algorithms may be used alongside this data for comprehensive localization.

7. CONCLUSION

7.1 Conclusion

The research presented in this paper demonstrates the viability and effectiveness of a computationally light audio localization method for robotics, leveraging rotating microphones to reduce hardware complexity and computational demands. By employing a rotational approach, the system achieves comparable accuracy to traditional Time Difference of Arrival (TDoA) methods while significantly minimizing the need for extensive microphone arrays and complex cross-correlation calculations. The integration of directional microphone enclosures, particularly the parabolic design, further enhances localization accuracy and extends the effective range of the system. Experimental results validate the robustness of the proposed algorithm modes—naive, averaging, and slicing—in various scenarios, highlighting their adaptability to different operational constraints and environmental conditions. The robot platform successfully navigates toward sound sources with minimal error, showcasing the practical applicability of the method in real-world robotics. This work not only addresses the challenges of computational efficiency and spatial constraints but also opens avenues for future research in optimizing enclosure designs and refining algorithmic performance for even greater precision. Overall, the proposed system represents a promising advancement in sound source localization, offering a scalable and efficient solution for small robots and swarms with limited processing power.

REFERENCES

[1] M. Yoshinari, M. Takahashi, and Y. Hioka, "Non-line-of-sight footstep localization for autonomous mobile robots," in *2024 10th International Conference on Mechatronics and Robotics Engineering (ICMRE)*, pp. 114–119, 2024.

[2] A. Badali, J.-M. Valin, F. Michaud, and P. Aarabi, "Evaluating real-time audio localization algorithms for artificial audition in robotics," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2033–2038, 2009.

[3] V. Kunin, M. Turqueti, J. Saniie, and E. Oruklu, "Direction of arrival estimation and localization using acoustic sensor arrays," *Journal of Sensor Technology*, vol. 1, September 2011.

[4] L. Kraljević, M. Russo, M. Stella, and M. Sikora, "Free-field tdoa-aoa sound source localization using three soundfield microphones," *IEEE Access*, vol. 8, pp. 87749–87761, 2020.

[5] H. Hu, M. Wang, M. Fu, and Y. Yang, "Sound source localization sensor of robot for tdoa method," in *2011 Third International Conference on Intelligent Human-Machine Systems and Cybernetics*, vol. 2, pp. 19–22, 2011.

[6] M. Luo, X. Chen, S. Cao, and X. Zhang, "Two new shrinking-circle methods for source localization based on tdoa measurementss," *Sensors*, vol. 4, September 2018.

[7] Z. P. Gabriel Jekateryńczuk, "A survey of sound source localization and detection methods and their applications," *PMC*, 2024.

[8] Z. A. Saad Benjelloun, "On sound dispersion and attenuation in simple and multi-fluids," in *Proceedings of meetings on acoustics*, December 2021. The Stokes-Kirchhoff relation expresses the sound attenuation as a function of the characteristics of the fluid, namely the density ρ, the dynamic viscosity, and other properties.

[9] O. Moreira-Tamayo, "ECEN 457 Lecture Slides," 2022. As of fall 2024.

[10] W. Kester, "What the nyquist criterion means to your sampled data system design," tech. rep., Analog Devices, Inc., 2009.

[11] M. A. Alessandro Carlini, Camille Bordeau, "Auditory localization: a comprehensive practical review," *Frontiers in Psychology*, 2024.

[12] Texas Instruments, "LM340, LM340A and LM7805 Family Wide VIN 1.5-A Fixed Voltage Regulators," 2023.

[13] Texas Instruments, "LM6132/LM6134 Dual and Quad Low Power 10 MHz Rail-to-Rail I/O Operational Amplifiers," 2014.

[14] Texas Instruments, "Filter Design Tool." https://webench.ti.com/filter-design-tool/.

[15] Texas Instruments, "OPAx354 250-MHz, Rail-to-Rail I/O, CMOS Operational Amplifiers," 2021.

[16] J. M. Fiore, *Operational Amplifiers and Linear Integrated Circuits*. Mohawk Valley Community College, 3rd ed., 2021.

[17] M. Achtenberg et al., "Review of peak signal detection methods in nanosecond pulses monitoring," *Measurement and Monitoring Systems*, 2019.

[18] N. G. Pavlidis et al., "High speed peak detection," *PNR Solution*, 2015.

[19] M. Fedkin and J. A, "Concentration with a parabolic reflector," *Penn State University*, 2023.

[20] R. C. Maher, "Parabolic dish microphone system," *Montana State University Technical Report*, 2005.

[21] FEETECH, "FS90R Micro Continuous Rotation Servo," 2013.

[22] Hadex, "The ULN2003 Driver Board," 2020.

[23] Kiatronics, "28BYJ-48 – 5V Stepper Motor," 2012.

[24] Arduino, "Arduino Uno Rev3 Datasheet," 2021.

[25] P. Glover and R. Bowtell, "On the definition of signal-to-noise ratio and contrast-to-noise ratio," *Concepts in Magnetic Resonance Part A*, vol. 32, no. 5, pp. 492–504, 2008.