KATHMANDU UNIVERSITY

SCHOOL OF ENGINEERING

DEPARTMENT OF ELECTRICAL & ELECTRONICS ENGINEERING

PROJECT PROGRESS REPORT

Binaural Recording

BY:

POLARJ SAPKOTA(31053)

MUKUL BHATTA(31031)

RISHAB RAWAL(31022)

SHREEJAN KARKI(31011)

SUBMITTED TO:

MR. MADHAV PRASAD PANDEY

DECEMBER 2021

ABSTRACT

The way we experience sounds through our ears in everyday life is one of the hardest things to reproduce on a digital system. Early stereo recording techniques managed to localize the angle of the sound sources but not the distance, which is crucial for recording sounds in the way humans hear it. The techniques developed to bypass this are arranged under the term 'Binaural Recording' which take a completely different approach to recording sounds. Modeling acoustic pressure as a function of several variables is currently the most widely employed and refined technique for such recording systems. This is the focus of this project.

ACKNOWLEDGEMENT

We would like to thank the whole Department of Electrical & Electronics Engineering for their continued support towards our academic endeavors. We would like to extend our sincere gratitude towards Dr. Samundra Gurung, Mr. Santosh Parajuli, Mr. Anil Lamichhane, Mr. Nawaraj Mahato, Dr. Shailendra Kumar Jha and Dr. Ram Kaji Budathoki who gave valuable advice in regard to choosing an academic project, which ultimately led us to realization of the true essence of an engineering project and choosing this project as the 3rd project of our undergraduate career. We would also like to thank Prof. Bhupendra Bimal Chhetri for providing us with insights on running experiments for the prototyping phase. We would like thank our project coordinator, Dr. Anup Thapa for reminding us of the value of teamwork on a semi-annual basis. Finally, we would like to thank our supervisor Mr. Madhav Prasad Pandey for his enthusiasm towards fundamentally guiding us to maintain an analytical approach towards engineering.

TABLE OF CONTENTS

ABSTRACT.....ii
ACKNOWLEDGEMENT.....iii
1. CHAPTER I: INTRODUCTION.....1
2. CHAPTER II: TECHNOLOGY AND LITERATURE SURVEY.....2
3. CHAPTER III: METHODOLOGY.....5
System Block Diagram.....6
Project Specifications.....7
Budget.....7
Limitations.....7
4. CHAPTER IV: SYSTEM ANALYSIS & PROGRESS SO FAR.....8
Project Timeline.....12
5. REFERENCES.....13
APPENDIX A.....14

LIST OF FIGURES

Figure 1: Front/Back Ambiguity [Source: Oculus VR Audio Design Guide]....2
Figure 2: Binaural Recording Block Diagram Representation....6
Figure 3: Plane Localization Scheme....8
Figure 4: 3D Localization Scheme with a 5-mic array....10
Figure 5: Microphone Amplifier Circuit....10
Figure 6: Polar Plot of Direction v/s Amplitude (Including Zeroes)....11
Figure 7: Polar Plot of Direction v/s Amplitude (Excluding Zeroes)....11

1. CHAPTER I: INTRODUCTION

Humans not only are capable of hearing sounds but also perceiving it. Our capabilities range from being able to accurately differentiate between different frequencies of sound to determining how loud or diffused a sound is. We can sense time delays between the original and the reflected sound to recognize echoes or reverberations. Using all these pieces of information collectively, we can almost accurately pinpoint the spatial location of the source of sound, which is called sound localization. This ability is facilitated by our enormous logical processing capabilities that perform processing tasks on sound such as: direction and frequency selective filtering, amplitude recognition, ITD (Inter-Aural Time-Difference), ILD (Inter-Aural Level-Difference). As evident by ears being one of the main five sensory organs, our hearing system is inherently complex and replication of such a mechanism requires deductions of the process to first principles.

Binaural recording aims to accomplish the same thing an ear can do with the use of an array of microphones, pre-amplifiers and other signals conditioning electronic circuits coupled with signal processing techniques, mainly DSP (Digital Signal Processing). Use-cases of binaural recording techniques generally encompass the audio-industry, film-industry and PC/Console games industry. VR (Virtual Reality) audio spatialization, where binaural sounds recorded at certain conditions are replicated through speakers or headphones to create a virtual 3D(3-Dimensional) soundscape in video games, is one of the major applications of such a recording system. Despite most applications existing largely in the entertainment-industry, sound localization has it's applications in a variety of places. Sound localization technology, which is a byproduct of binaural recording is useful in accurately mapping out surfaces with a high level of accuracy depending on the accuracy of the used localization techniques. These are also used in special underwater communication-devices known as Acoustic Modems which can be used to map underwater surfaces or to establish communication with UUV(Unmanned Underwater Vehicles). Other two common applications are; recording songs in order to create a concert or studio-like experience for the viewer & sampling of sounds for reproduction in movie scores which is famously done for movies by companies such as Dolby Laboratories.

2. CHAPTER II: TECHNOLOGY AND LITERATURE SURVEY

Localization, most-widely and most successfully is done by determination of several variables as a function of Acoustic Pressure. This function of acoustic pressure is known as an HRTF, which has gained much fame in audio. HRTFs can be modelled as the following five-variable functions:

$$H_L = H_L(r, \theta, \varphi, \omega, \alpha) = P_L(r, \theta, \varphi, \omega, \alpha) / P_0(r, \omega)$$

$$H_R = H_R(r, \theta, \varphi, \omega, \alpha) = P_R(r, \theta, \varphi, \omega, \alpha) / P_0(r, \omega)$$

Where H_L & H_R represent respective HRTFs of the left and right ear, PL & PR represent amplitude of sound pressure waves at the entrance of both left and right ear canals, P0 is the amplitude of the pressure wave at the centre of the head with the listener removed, L is left ear, R is right ear, r is distance between source and centre of the head, θ is source angular position, φ is elevation angle, ω is angular velocity, α is equivalent dimension of the head.

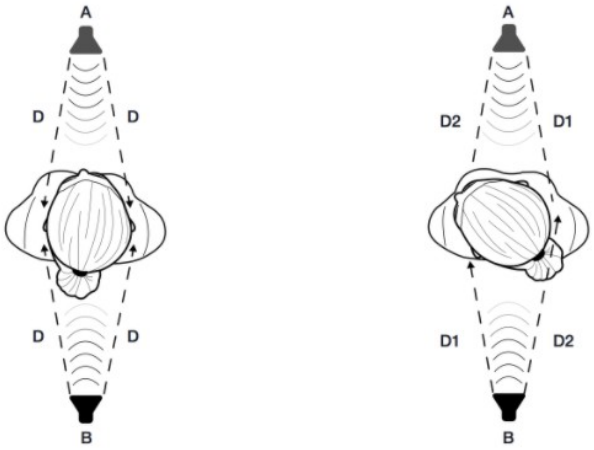

Figure 1: Front/Back Ambiguity [Source: Oculus VR Audio Design Guide]

The figure above presents a problem that is resolved easily by the human ear. It shows two sound source, one at the back & one at the front, equidistant from the listener. Even when nonlinearity of the environment is taken into consideration, our ears will have a hard time localizing the sources of sound, because sound arrives at both ears simultaneously. How does the listener determine where one source is located? This perplexing problem has a simple solution, if the listener simply turns their head horizontally, it creates a time-difference between the arrival time of different sources which then simplifies the problem and sound is localized easily. This simple action of turning the head to differentiate between the location of two ambiguous sources is executed using multiple microphones i.e. a microphone array. The microphone array is arranged such that each microphone sits at different variable heights and a fixed planar distance from each other. This allows only one diagonal plane of symmetry to exist in the whole 3D space where the ambiguity presented by Figure 1. occurs and if it

were to occur, the array can be controlled such that whenever the ITD approaches zero, the microphones change their height by a small amount, rotating the plane of symmetry to a convenient location. Ambiguities as such of the above example are even more easily resolved when the HRTFs depend on variables other than the ITD as well. Further ambiguities such as motion parallax and loudness are also resolved by HRTFs.

With these basics in mind, the following other techniques are use to localize sound precisely:

1. Loudness - The louder the sound, the nearer it seems, the lesser the loudness the further we assume it to be. Example: music playing with low volume in the same room is perceived to be playing in another room or another house
2. Initial time delay - Interval between the direct sound and it's first reflection
3. Ratio of direct sound to reverberant sound - When direct sound/reverb sound is small, the sound is assumed to be closer. When it is bigger, the sound is perceived to be farther.
4. Motion parallax - Sounds with small ITD are perceived to be nearer. Example of a loud superbike passing by a road 300m away seems to be nearer to us than it actually is. Another example is the airplane taking off/landing near an airport
5. HF Attenuation - HF sounds attenuate faster

How to create HRTFs?

Huge HRTF datasets are available online for use in prototyping. Creation of HRTFs require a mold in the shape of a human ear and a mannequin head closely resembling human head. After these tasks, obtaining a HRTF is a matter of making precise measurements in a controlled environment, interpolating, passing test signals, iterating and perfecting. Devices well isolated from environmental effects prevent interfering HRTFs from affecting the sound experience. Bluetooth communication is not recommended while using listening devices as latency of up to 500ms is introduced.

Environmental Modeling

The 'shoebox' model can be used for environmental modeling. This model places the test subject inside a room to allow modeling of environmental effects caused by virtue of geometry such as reverb, early reflections and such. Environmental Modeling techniques for VR as the basis of precise HRTFs can be implemented in the following ways:

- Using volumetric source-based modeling
- Introducing the Doppler effect by increasing the freq. as sound source approaches and decrease the freq. as it leaves
- Delaying the time of arrival of the sound can seem intuitive in some cases such as the case when recording a thunder. We don't immediately hear the sound of the thunder after observing it. But popular media has made it the intuitive to most people that even long distance events are immediately audible.

- Avoid head-locked audio. Most music is recorded in stereo mixes and even VR applications use stereo playback devices but stereo doesn't spatialize the sound. It pans sound to a specific frequency so that only one ear perceives it well. But imagine the same thing in a game. If you hear birds chirping from a certain direction, when you turn your head, you expect the bird's chirp to stay in the same location, not rotate along with your head. This kills immersion. So mixing with ambisonics (spherical 3D sound field) gives better results compared to stereo when accurate spatialization of the localization is required.
- Make sure to keep latency under 100ms at all times.
- Other effects such as low-pass filtering can emulated sounds heard in an underwater setting as HFs attenuate rather quickly underwater

3. CHAPTER III: METHODOLOGY

- Thoroughly read through and document contents of accessible literature.
- Search for a well-documented VR company website and go through available articles.
- Go through review papers on the topic and summarize.
- Search for similar projects and research papers on the topic, read through and summarize.
- Determine the best possible approach considering available tools and resources.
- Study about common architectures, bit resolution & sampling rates needed to decide the right ADC (Analog-to-Digital Converter).
- Develop the system block diagram.
- Purchase mannequin dummy head
- Determine the HRTF of the dummy in a minimum-noise environment.
- Model multiple iterations of required HRTF in a recording studio.
- Build a small chamber with uniform shape and smooth geometry.
- Use basic soundproofing/reverb cancellation techniques & materials for the chamber.
- Calculate best location for placement of a high quality speaker within the chamber.
- Troubleshoot with test signals for the best impulse response
- Process & condition the localized audio signals with MATLAB or Python's DSP tools
- Fetch the recorded sounds directly to a laptop in '.wav' or other uncompressed lossless formats during testing phases.
- Design required digital filters, mixers, etc and troubleshoot to make mic array capable of extracting certain sounds, such as isolating and recording ambient sounds.
- Implement the completed signal processing algorithms on a single board computer.
- Design a dummy-replica of the human head with mic array as the ear.
- Authorize & gain permission for data collection.
- Collect inter-aural length data from willing volunteers.
- Organize data and determine the best fit excluding outliers.
- Compare the collected data with HRTF datasets from laboratories around the world which includes the IRCAM Listen [1], MIT KEMAR [2] & CPIC HRTF [3] databases and document the shortcomings & advantages of the collected data.

- Design at least two dummy heads according to the most feasible specifications covering two largest groups of people using non-reflective acoustic materials.
- Store the final recording in compressed form.
- Store the final recording in lossless '.wav' or '.flac' format.
- Write a script to run a third-party conversion program to compress the '.wav' file to '.mp3' or '.aac' formats.

System Block Diagram

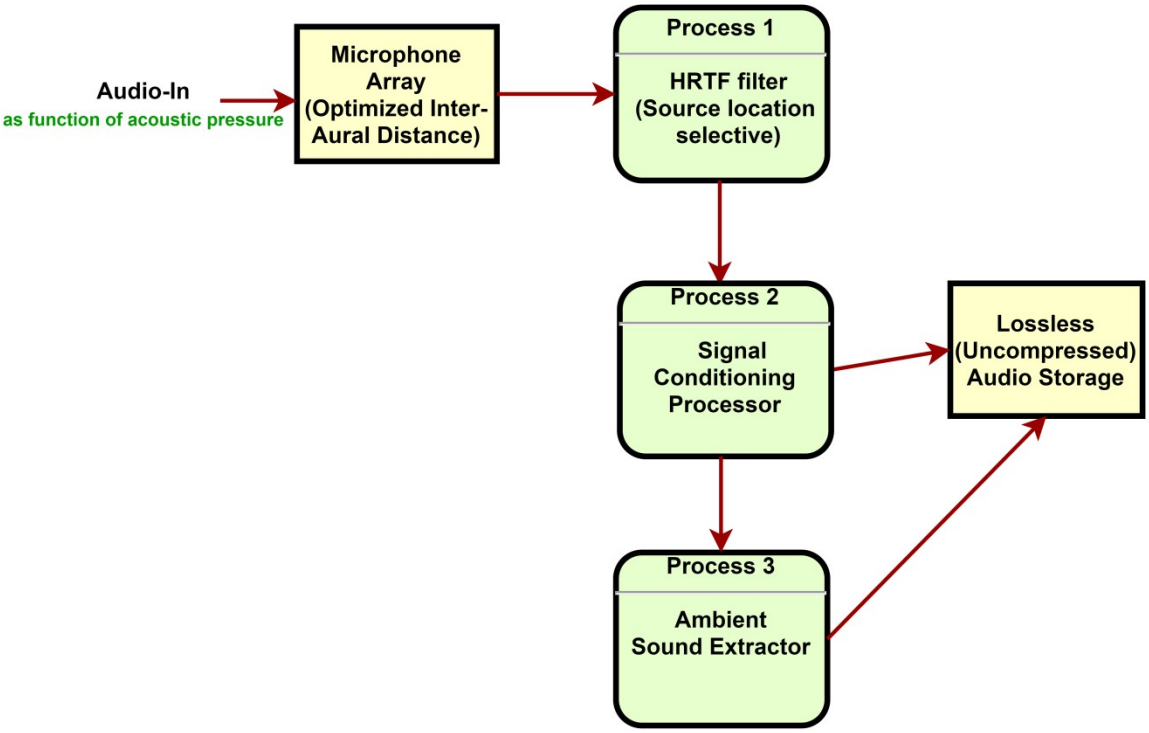

Figure 2: Binaural Recording System Block Diagram Representation

Project Specifications

Materials & Components:

- Acoustic Energy Absorbing Material
- ADCs with minimum resolution of 10-bits
- A Single-Board Computer: Asus Tinkerboard, Raspberry Pi, Arduino Mega 2560 etc.
- Five microphones with good frequency response for the range 50Hz - 18kHz
- Rubber Molding tools and materials
- Mannequin Head

Min. dimensions of the 'Shoe-box model': 1m x 0.75m x 0.5m

Budget

Microphones - Rs. 2000

Mannequin Head - Rs. 100

Acoustic Materials - Rs. 400

Other materials - Rs. 600

Single Board Computer - Rs. 2000

Total Estimated Budget: Rs. 5100

Note: All costs are approximations at best

Limitations

Most of the obvious limitations are expected to be brought forward by the damping capabilities of the inexpensive acoustic materials planned to be used. Other subtle limitations include, the ready-made mannequin head's dimensions not resembling a real person's head with the best accuracy even after modifications. Microphones to be used also are considerably inexpensive compared to studio quality condenser microphones. As such, the system won't be able to process sounds with frequencies above 18kHz. Prototype processing on a computer does not present any limitations although the final system implementation on single board computers might present some unforeseen challenges.

4. CHAPTER IV: SYSTEM ANALYSIS & PROGRESS

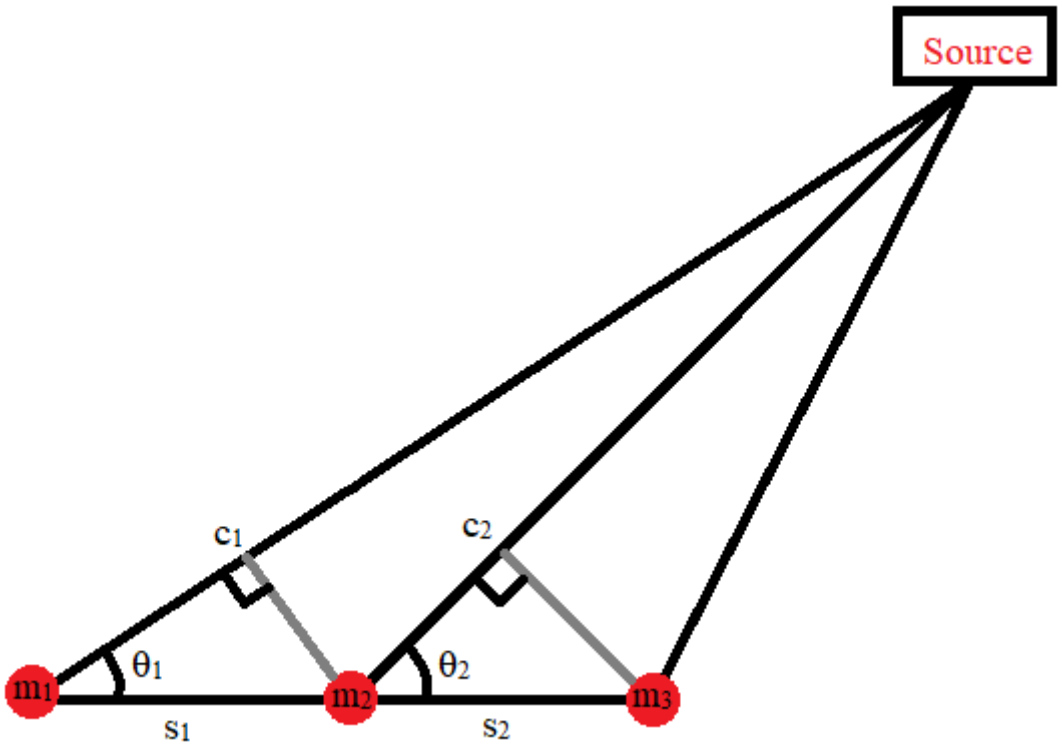

Figure 3: Plane Localization Scheme

For localization of a sound source located in any arbitrary position either to the left of m_1 or to the right of m_3, consider the fact that it either strikes m_3 or m_1 first among the three mics.[ǂ]

Consider the approximation that waves travelling from a point source look like a plane wavefront the further we move from it. This allows us to assume that if a sound source is located to the right of m_3, the pressure wave makes contact with m_3 first, thus inducing a voltage on it. Let the distance to the sound source from m_2 be denoted by *S*.

We define this exact moment as $t_0=0$. The plane wavefront assumption allows us to draw a line (t=0) perpendicular to the path our wavefront takes on its way to m_2, at the point c_2. This line indicates the starting point of the master clock. The moment our wavefront makes contact with m_2 can be stamped as t_1. Knowing t_1 & t_0 and the fact that velocity of sound in air ~332 m/s, the distance between m_2 & c_2 can be approximated as

$$c_2 - m_2 = 332 \cdot (t_1 - t_0)$$

Similarly, for m_3, the distance between m_1 & c_1 can be approximated as

$$c_1 - m_1 = 332 \cdot (t_2 - t_1)$$

[ǂ]The assumption won't work when m_3 is closest to the sound source (Front/Back Ambiguity)

Knowing this, from basic trig identities & with the distances between mics (s_1 & s_2) known, the angles θ_1 & θ_2 are given approximately by:

$$\cos \theta_1 = (c_1 - m_1) \div s_1$$

$$\cos \theta_2 = (c_2 - m_2) \div s_2$$

Therefore, the angles are:

$$\theta_1 = \cos^{-1} [\, (c_1 - m_1) \div s_1 \,]$$

$$\theta_2 = \cos^{-1} [\, (c_2 - m_2) \div s_2 \,]$$

Let the path of sound from Source to m_2 be denoted by S_2 & from S to m_3 be denoted by S_3

Then equations of the lines S_2 & S_3 are given by:

$$x_2 = x_1 \cdot \tan \theta_1 + (t_1 - t_0)$$

$$x_2 = x_1 \cdot \tan \theta_2 + (t_2 - t_1)$$

Which can be expressed as a matrix equation:

$$\begin{bmatrix} \tan \theta 1 & -1 \\ \tan \theta 2 & -1 \end{bmatrix} \cdot \begin{bmatrix} x1 \\ x2 \end{bmatrix} = \begin{bmatrix} -t1 + t0 \\ -t2 + t1 \end{bmatrix}$$

Similarly for a vertical mic arrangement, the equations are:

$$\begin{bmatrix} \tan \theta 3 & -1 \\ \tan \theta 4 & -1 \end{bmatrix} \cdot \begin{bmatrix} x3 \\ x4 \end{bmatrix} = \begin{bmatrix} -t3 + t2 \\ -t4 + t3 \end{bmatrix}$$

Solving for the x's in each equation gives the resulting co-ordinate points in a plane, form which the azimuth and plane angles $\boldsymbol{\theta}$ & $\boldsymbol{\phi}$ which is the approximate location of the sound source. Extending the same process for a vertical arrangement and solving for those corresponding x's gives two more co-ordinate points which further localizes the sound source in that plane. Combining those two points to create three coordinate points result in a three co-ordinate points which is the approximate location of the source in 3D space. Location of the sound source can be determined in two planes by keeping the mics in the following configuration:

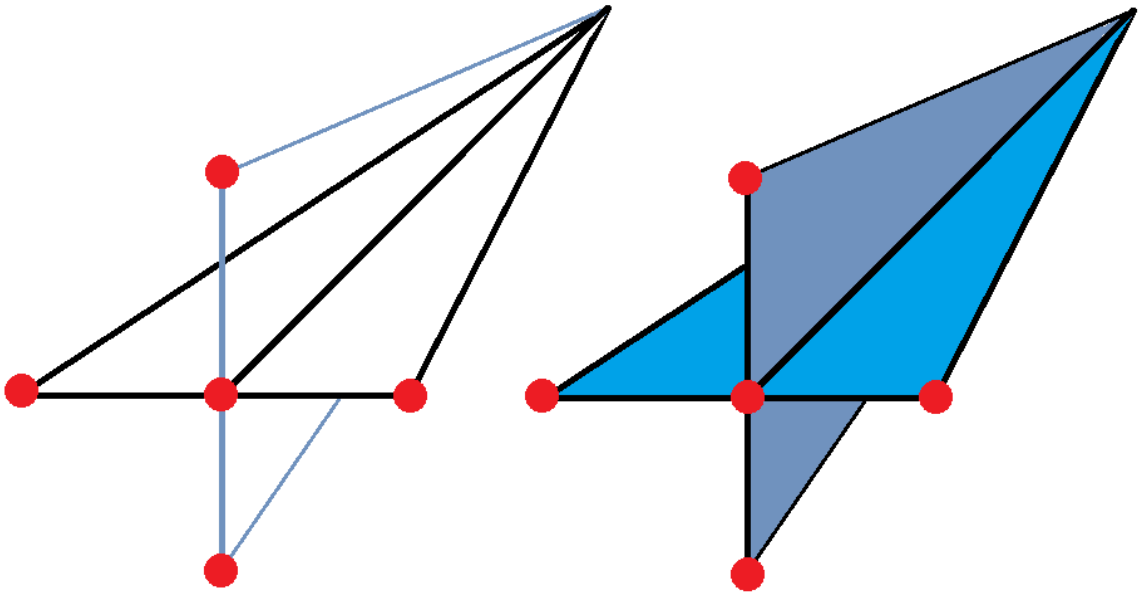

Figure 4: 3D Localization Scheme with a 5-mic array

For practical considerations of localizing the sound source, tests were conducted on the available condenser microphones available in the lab & in earphones. Mics salvaged from earphones were found to have better frequency response & noise immunity. The plane sound field of the mic was recorded in a moderately silent environment with some reverb. The simple microphone pre-amp circuit in *Figure 5* was placed on top of a servo motor controlled mini-rig. Data was logged into a '.csv' file using PuTTY (An open-source terminal emulator, serial console & network file transfer application), which was used to access serial communication port of the PC (COM ports). Data was received into the serial port through a simple script for the Arduino prototype board. The code can be found in the Appendix.

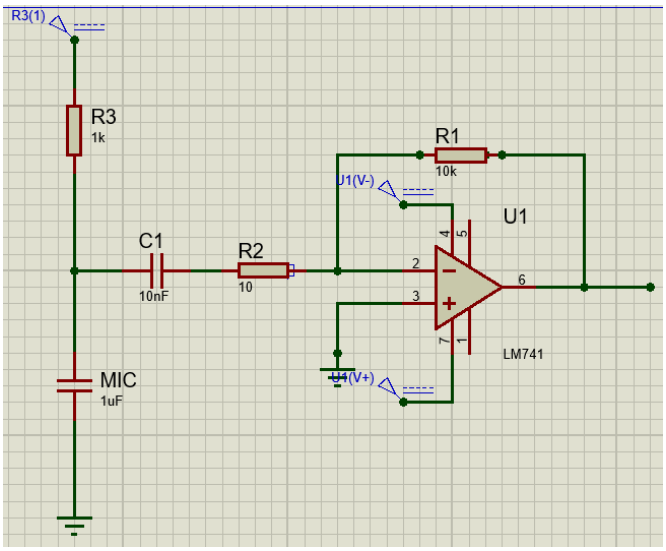

Figure 5: Microphone Amplifier Circuit

The received data was plotted as a polar plot. The servo was programmed to move in steps of 1º with each iteration. Data was logged before stepping onto the iteration for the next degree.

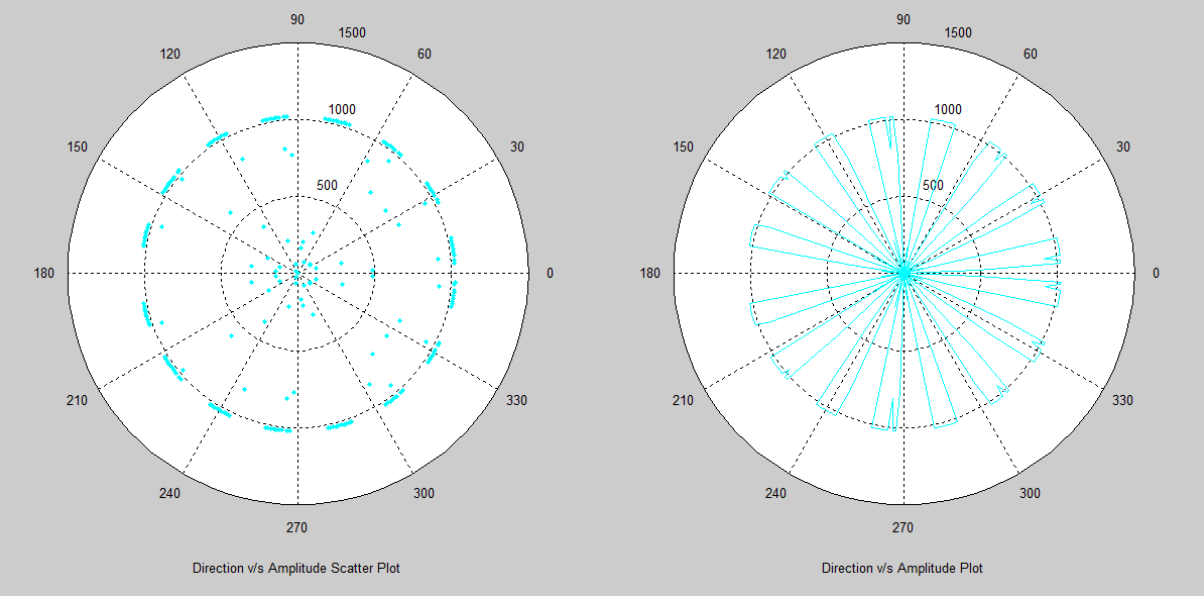

Figure 6: Polar Plot of Direction v/s Amplitude (Including Zeroes)

The plots reveal that the mic picks up sounds from only half the directions. This was first considered as a measurement error since there were a lot of zeroes in the logged data. Removing them and plotting them again as shown in *Figure 7* confirmed that this was indeed the actual behavior. Amplitudes observed are for a sine wave played at 1178 Hz from a smartphone speaker which was placed approximately 3 cm from the mic.

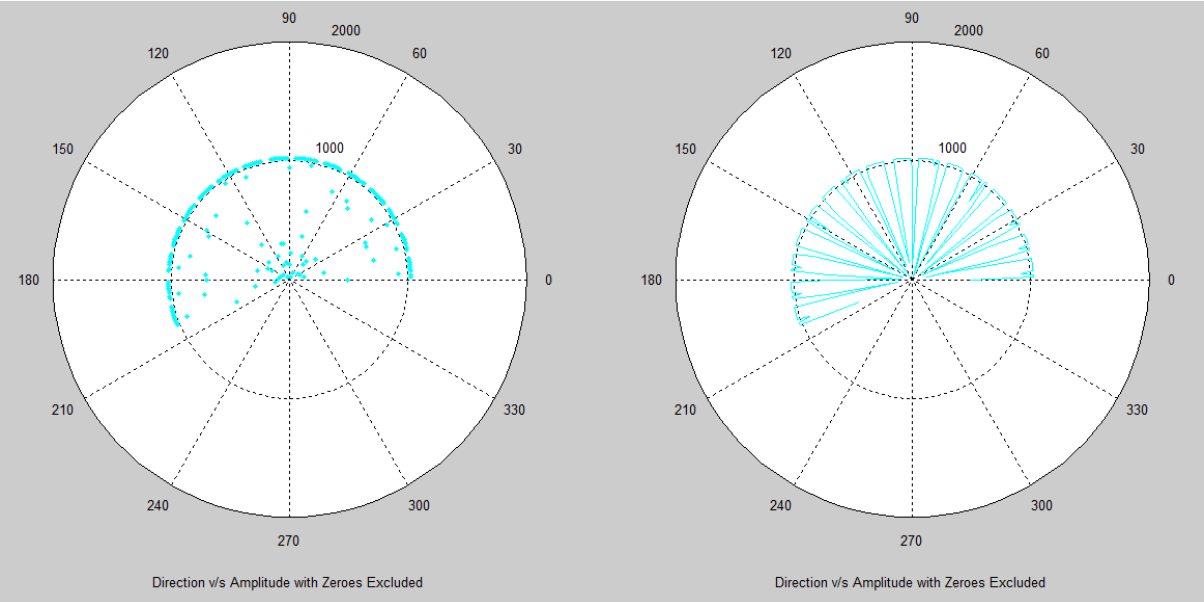

Figure 7: Polar Plot of Direction v/s Amplitude (Excluding Zeroes)

Project Timeline

Tasks	Mar/Apr	May/Jun	Jul/Aug	Sep/Oct	Nov/Dec	Jan/Feb
Literature Review						
Proposal Submission						
Prototype Design						
Mid-Term Report						
Final System Synthesis						
Final Report and Demo						

Completed Task

Remaining Task

5. REFERENCES

1 *Listen HRTF Database,* L'*Ircam*, *Institut de Recherche et Coordination Acoustique/Musique, Sept. 2002. [Online]. Available:* http://recherche.ircam.fr/equipes/salles/listen/

1 B. Gardner, K. Martin, "HRTF Measurements of a KEMAR Dummy-Head Microphone ", MIT Media Lab, Cambridge, Massachusetts, USA, Technical Report #280, May 1994, [Online]. Available: https://sound.media.mit.edu/resources/KEMAR.html

2 V. R. Algazi, R. O. Duda, D. M. Thompson and C. Avendano, "The CIPIC HRTF Database," Proc. 2001 IEEE Workshop on Applications of Signal Processing to Audio and Electroacoustics, pp. 99-102, Mohonk Mountain House, New Paltz, NY, Oct. 21-24, 2001, [Online]. Available: https://www.ece.ucdavis.edu/cipic/spatial-sound/hrtf-data/

3 Wen Zhang , Parasanga N. Samarasinghe , Hanchi Chen and Thushara D. Abhayapala., "Surround by sound: A review of spatial Audio Recording and Reproduction," in MDPI applied sciences, May.2017. (Online). Available: https://www.mdpi.com/2076-3417/7/5/532.

4 Philip Coleman , Andreas Franck , Jon Francombe , Qingju Liu , Teofilo de Campos, Richard J. Hughes, Dylan Menzies , Marcos F. Simon G ´ alvez, Yan Tang ´ , James Woodcock, Philip J. B. Jackson, Frank Melchior, Chris Pike , Filippo Maria Fazi, Trevor J. Cox, and Adrian Hilton, "An Audio-Visual System for Object-Based Audio: From Recording to Listening," in IEEE Transactions On Multimedia" ,Aug . 2018 . (Online). Available: https://ieeexplore.ieee.org/document/8260969

5 Natsuki Ueno , Student Member, IEEE, Shoichi Koyama., Member, IEEE., "Sound Field Recording Using Distributed Microphones Based on Harmonic Analysis of Infinite Order,"in in IEEE Transactions On Multimedia", Jan.2018 .(Online). Available: http://www.ieee.org/publications standards/publications/rights/index.html.

6 Kazuhiro Iida., *Head-Related Transfer Function and Acoustic Virtual Reality* . Narashino, Chiba, Japan

7 Bosun Xie., *Head-Related Transfer Function and Virtual Auditory Display.*

8 Brad Osgood., The Fourier Transform and its Applications (Online). https://see.stanford.edu/course/ee261.

APPENDIX A

```
#include <Servo.h>
bool EndOfMeasurement = false;
int mic=A0;
int amplitude[361];
Servo myservo;
int pos = 0;    // variable to store the servo position
void setup() {
        myservo.attach(6);  // attaches the servo on pin 6 to the servo object
        pinMode(mic, INPUT);
        Serial.begin(9600);}
void loop() {
        for (pos = 0; pos <= 180; pos += 1) {      // goes from 0 degrees to 180 degrees
                myservo.write(pos);                 // tell servo to go to position in variable 'pos'
                delay(100);                         // waits 15ms for the servo to reach the position
                amplitude[pos]=analogRead(mic);
                Serial.println(amplitude[pos]);
                delay(100);}
        for (pos = 180; pos >= 0; pos -= 1) {   // goes from 180 degrees to 0 degrees
                myservo.write(pos);              // tell servo to go to position in variable 'pos'
                delay(100);                      // waits 15ms for the servo to reach the position
                amplitude[361-pos]=analogRead(mic);
                Serial.println(amplitude[pos]);
                delay(100); } }
```