Earbeamer: A Parallel Beamforming Hearing Aid System

Niket Gupta, CSE, Aaron Lucia, CSE, Nathan Dunn, CSE, and Matteo Puzella, EE

Abstract—Earbeamer is stationary, wall-mounted hearing aid system targeted at the senior citizen population that allows users precise control over the volume of particular individuals within the room. By applying beamforming in parallel over a microphone array, the audio of each identified individual is isolated, and may be attenuated or amplified. Through an Xbox Kinect, the movements of each individual are tracked, ensuring that a conversation is unimpeded regardless of movement within the room.

Index Terms—Beamforming, Microphone Arrays, Acoustics, Hearing Loss, Signal Processing

1 Introduction

HEARING loss is a common problem among the senior citizen population. As we get older, parts of the inner ear that are sensitive to sound begin to atrophy a process that is often exacerbated by many factors that are commonly found among the elderly, such as diabetes, high blood pressure, and even some chemotherapy drugs. Presbycusis age related hearing loss affects about 1 in 3 Americans over the age of 65 [6]. By age 75, this number increases to about 1 in 2.

Its prevalence is concerning, as adequate hearing is a vital requirement for communication. The typical onset of presbycusis coincides with many major social changes in the life of an individual. An individual may be facing retirement, or losing mobility due to age-related ailments.

The loss of these social interactions can compound with hearing loss and have profound effects on cognition. In a study of 2,304 adults with individuals with hearing impairments, those without hearing assistance were 50% more likely to suffer from depression [7]. A separate study found that dementia progressed more quickly among the hearing impaired population than a healthy population, with cognitive performance declining 30 40% faster over an equal period of time [3].

1.1 Existing Solutions

The current hearing aid market provides hearing aids that use beamforming. The beamforming hearing aid consists of multiple omnidirectional microphones that form a beam signal. It helps attenuate background noise while focusing toward the target sound. The big issue with the beamforming hearing aids is that the sensitivity of the hearing aid drops off at low frequencies (1000 Hz). In order for the beamforming hearing aid to accommodate this situation, additional gain must be implemented in the hearing aid to hear low frequencies. However, additional gain comes at the cost of internal noise in the hearing aid, which is unwanted.

A common polar amplification pattern for a simple beamforming hearing aid is a cardioid. By delaying microphone outputs from the sides and rear of the hearing aid, sounds arriving from the those directions can be attenuated, while sounds arriving from directly in front of the user are amplified.

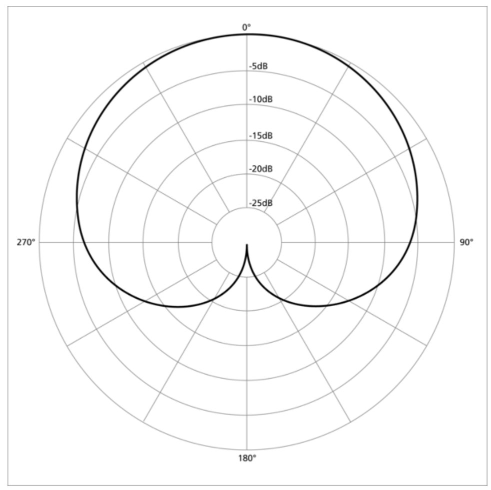


Fig. 1. Cardioid Pattern response for directional hearing aids. 90^o represents the area directly in front of the user

However, these systems have a common flaw: they do not distinguish between sound sources when amplifying sound. When a hearing aid user is attempting to listen to a conversation, anything that is not part of that conversation is background noise, whether that be the television in the room, or a separate conversation across the room. Hearing aids without a form of source isolation will amplify wanted sources, as well as unwanted sources, creating background noise. This is undesirable, as Kochkin [1] found that a quarter of individuals who own hearing aids but do not use them cite poor performance amid background noise as the primary reason.

Further, as is shown by the polar plot of the directional hearing aid in Figure 1, many current directional solutions are dependent upon the listener physically looking at a target to obtain maximum amplification. However, this is often not the case in an actual conversation. There are many instances where a participant in a conversation may not be actively looking at other participants, such as when they are looking at a television, or moving through the room. In these scenarios, a listeners ability to perceive the conversation should not be hindered by head position.

1.2 Requirements

Noting the two above scenarios, we have proposed a hearing aid system that allows an elderly user:

1) To selectively attenuate or amplify nearby human targets within an indoor space, -essentially giving the user the ability to mute or turn the volume up on individuals within the room.
2) To move about the room without negatively affecting the amplification of their conversation

We achieve both of these goals using beamforming through a stationary array of microphones within the room, a process that we will describe in detail in Section II.

1.3 Specifications

The specifications for the hearing aid system are summarized in Table 1:

Specification	Value
Array Width	< 2m
Target Identification Range	> 20ft
Angle of Operation	30^o to 150^o
Maximum Number of Targets	> 4
Beamwidth	$< 15^o$
Bandwidth	600 2400 Hz
Delay	< 300ms

TABLE 1
Specifications for Device

Size and Range of Device: To formulate our specifications, we made the assumption that the system will be operating within a users home in a space reserved for entertaining guests, such as a family or living room. Considering a 20 x 20 living room, this assumption provides the maximum distance that the system must identify and listen to targets, as well as the maximum allowed size of the system.

Within the relatively small space of a living room, it is important to ensure that any potential device does not disrupt the normal daily life of any occupants. We desire a wall-mounted system to leave as much floor space for the occupant as possible. The system should also not overtly draw attention to itself and dominate the room. To accomplish this, we set a device size limit of 2 meters in length, about the size of a large hanging piece of artwork.

Beamwidth: To effectively isolate the output of sound sources, we must be able to encapsulate each target within distinct, non-overlapping beams, as seen in Figure 2 If each target is centered within a beam, then the 3dB beamwidth of that beam must not intrude into the 3dB beamwidth of another beam. We considered a typical living room couch seat as the minimally spaced placement that any two people will be arranged within the room, as seen in Figure 2. If a typical couch cushion is 25 in width, and the couch is 8 from the wall, then the minimum beamwidth is approximately 15^o

Bandwidth: In typical telephony applications, the transmitted human speech spectrum range is about 300 3300 Hz5. in order to maintain intelligibility. However, Section II will show that beamwidth may be gained by sacrificing bandwidth, so we limit ourselves to 600 2400 Hz. This is similar to the 300 2700 Hz bandwidth used for long-distance telephone connections in the 1980s [2].

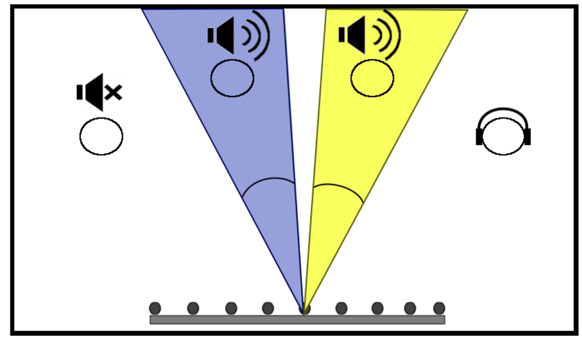


Fig. 2. Desired behavior for our system. A wall mounted device is able to selectively listen to multiple targets within the room for a listener, here wearing headphones

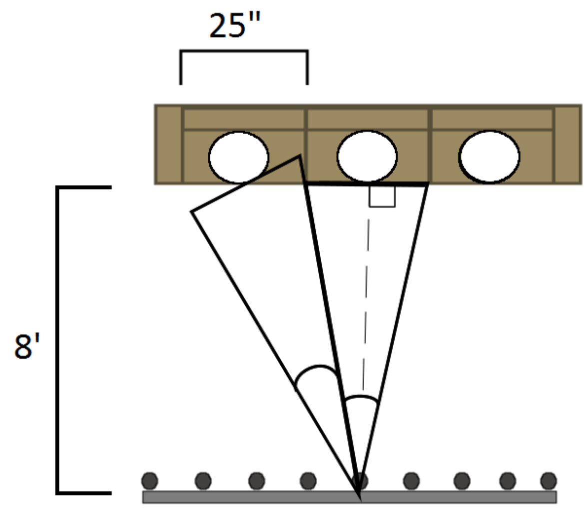


Fig. 3. Finding the minimum beamwidth needed to encapsulate targets within individual beams

Delay: To provide seamless conversations, the delay between reception of sound and playback to the user must be minimal. The ITU G-177 specification for VoIP mandates a two-way delay of less than 300 ms, so we used this number to provide an upper bound on delay.

Angle of Operation:

For a wall mounted system, microphones must be able to target and listen over a wide range of angles to provide adequate coverage for the entire room. For a microphone array parallel to a wall, we would ideally want to isolate and amplify sound over a range of 180^o. However, the nature of our signal processing method, beamforming, ensures that isolating sound from sources close to the extreme ends of this range is difficult (at $0^o/180^o$, ie. when a source is close to the same wall where the system is mounted).

For reasons that will be expanded upon in Section 2.2 we cannot amplify targets at 180^o without also amplifying sound at 0^o. This is called an "endfire" array response , and

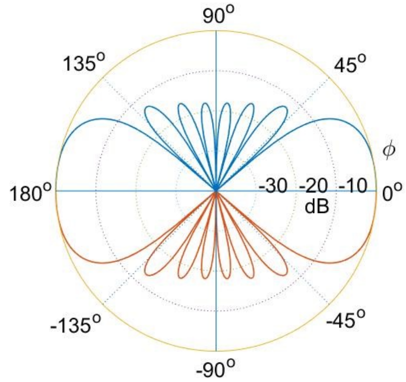


Fig. 4. This is the array response when the elements are steered toward 180^o or 0^o. You cannot amplify one direction without also amplifying the other direction

may be seen in Figure 4.

We desire for the user to have individual control over each possible source of sound, so we have chosen to limit the angle of operation to 30^o to 150^o to avoid the endfire configuration. For our problem, this is a reasonable limitation, as most targets will be present at some point within the room, not hugging the wall of the device.

2 Design

2.1 Overview

Our approach to addressing this problem harnesses the power of audio beam-forming and mates it with an intuitive user interface. This is enabled by a visual tracking system alongside a mobile app that allows intuitive user interaction for system control. The beam-forming approach has been used several times before and is well documented. Furthermore, our processing algorithm gives us a very low latency so that this system viable for real-time communication.

Our aim was to automate the system as much as possible while allowing the user to control only what was relevant, specifically which individuals they would like to isolate and hear. The Microsoft Kinect plays a key role in allowing this automation with its ability to identify and map unique individuals. The mobile app displays this data and allows for user selection. The software program performing the beam-forming processes the audio streams from the beam-forming array and outputs them to the users headset. An overview of this system is shown in Figure 3.

2.2 Microphone Array

The microphone array is the method through which sound is sampled spatially from the environment. Through classes such as ECE 313 and ECE 563, we are familiar with how an analog signal may be sampled in time by a set sampling period. An array of elements displaced by a set distance d samples the same signal with a relative phase shift between elements. By adding the output of each element together, signals from certain directions are added constructively, while signals from other directions are added destructively.

An array of elements with identical radiation patterns can be described by a term called the array factor, which for a one-dimensional linear array of n elements can be written as:

$$A(\phi) = a_0 + a_1 e^{jkd\cos(\phi)} + ... + a_n e^{jnkd\cos(\phi)}$$
$$A(\phi) = \Sigma_n a_n e^{jnkd\cos(\phi)} \quad (1)$$

Where d is the distance between sampling elements, k is the wavenumber of an incoming wave, and a_o a_n are complex coefficients [8]. This sum of complex exponentials completely describes the geometry of the array, with each term representing the relative phase shift resulting from the time that it takes for a wave to propagate from element to element.

The array factor has a dramatic effect on the directivity of the array. For a wave incoming at a direction of ϕ, if each element has an identical power gain of $G(\phi)$, then the gain of the entire array system $G_{tot}(\phi)$ is [8]:

$$G_{tot}(\phi) = |A(\phi)|^2 G(\phi) \quad (2)$$

For example, Figure 6 contains a polar plot of the term $|A(\phi)|^2$ for a linear array of 8 elements, with coefficients $a_0 = \ldots = a8 = 1/8$, meaning that each element is equally weighted. In this scenario, the maximum power gain occurs when a wave is arriving perpendicular to the linear array, at 90^o, also known as broadside. Intuitively, this is the direction where all microphone inputs add together in-phase.

By changing the coefficients $a_0 \ldots a_n$ to a set of complex exponentials, each sampling element provides a phase shift (i.e. a time delay) to the signal that it is sampling. The direction of the main beam in the polar plot of $|A(\phi)|^2$ can be translated to another direction, called the steering angle. Figure 5 displays the array pattern for a beam aimed 35^o from broadside.

From Equation 2, we can see that there are two terms affecting the power gain polar pattern of our linear array:

- $G(\phi)$ determined by the microphone selection
- $|A(\phi)|^2$ determined by the geometry of the microphone elements

By optimizing both of these terms, we can minimize the beamwidth of the array, and meet our specifications

2.2.1 Microphones

For microphone selection, we had access to the 16 omnidirectional ADMP510 MEMS microphones used in the SDP 16 beamforming project. As the SDP 16 team noted, this particular model of microphone has a relatively linear frequency response within the frequency band targeted by our system [5].

Given this property of the microphones, we decided to use SDP16s microphones within our own design, to ensure that all frequencies within the targeted band are amplified equally. However, as we are receiving these microphones used, we will need to complete a verification procedure on each microphone, to ensure that it is still functioning after storage. To accomplish this calibration procedure, we will record low, medium and high frequency tones on each

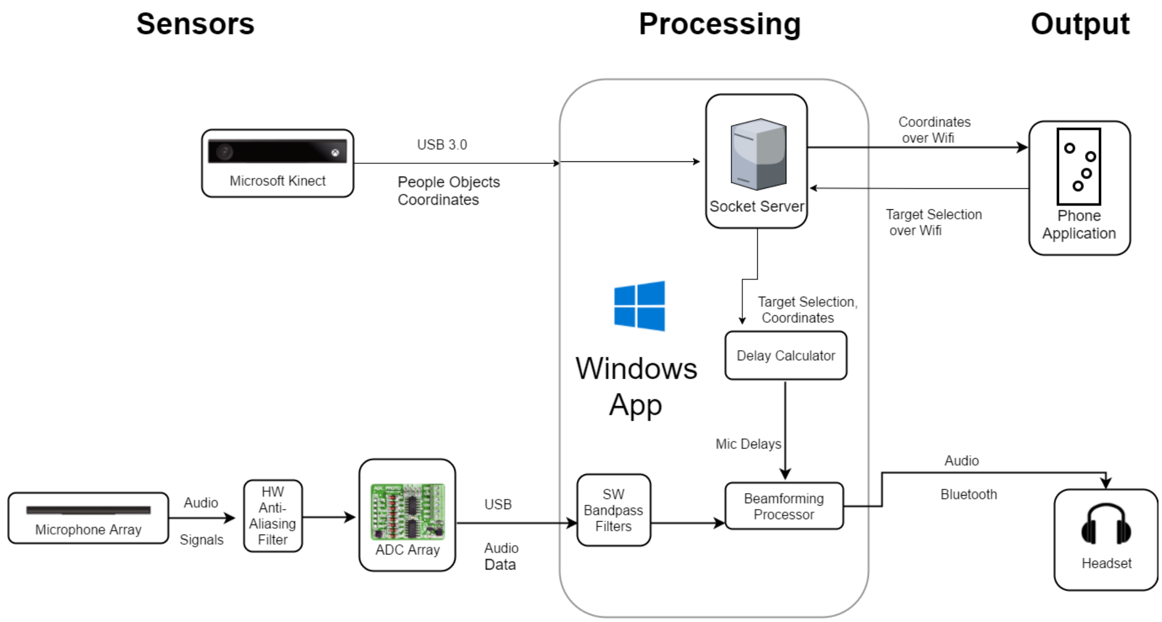


Fig. 5. Caption

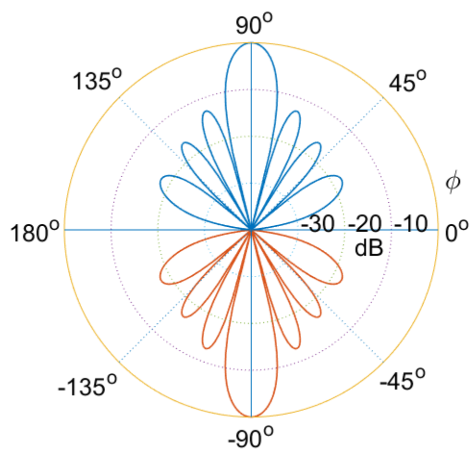


Fig. 6. Array Power gain $|A(\phi)|^2$ for an array of 8 elements, spaced one-half a wavelength apart, with each microphone equally weighted

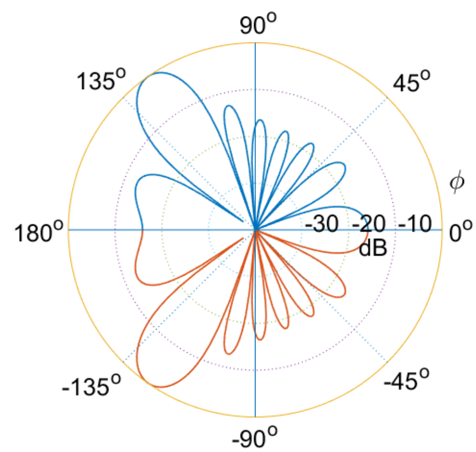


Fig. 7. Array Power gain $|A(\phi)|^2$ for an array of 8 elements, steered towards 125^o

microphone, and then play each tone back. If the playback tone matches the original tone, then we can verify the microphone as functional.

Note that these microphones are omnidirectional, so tones are uniformly amplified in terms of direction. Thus, Equation 2 simplifies to:

$$G_{tot}(\phi) = |A(\phi)|^2 \tag{3}$$

2.2.2 Array Geometry

As Equation 3 shows, the array factor is the only term that determines the directivity of the array. Therefore, in order to optimize the beamwidth, we need to select an optimal microphone geometry.

Orfanidis shows that for a uniform linear array, the 3dB beamwidth may be approximated as [8]:

$$\Delta 3dB = \frac{0.886\lambda}{\sin(\phi_0)Nd}b \tag{4}$$

Where ϕ_0 is the steering angle, λ is the wavelength of tone, N is the number of microphones, d is the microphone distance, and b is a factor dependent on the weighting applied to each microphone.

Equation 4 shows that beamwidth will increase as the beam is steered towards 0 or 180o, and as frequency decreases. This creates an issue for our beamformer, as it means that different frequencies within the human speech spectrum will produce different beamwidths.

Increasing d or N will decrease beamwidth, but the distance between microphones cannot be increased beyond λL/2, where λL is the largest wavelength within the targeted frequency band. This is the Nyquist criteria for spa-

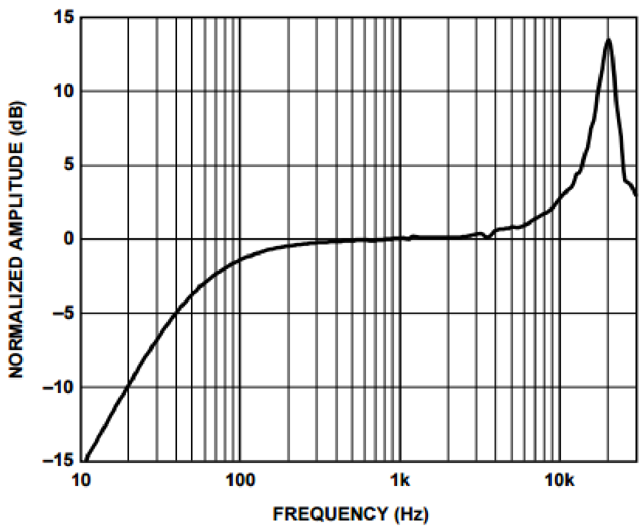


Fig. 8. Frequency response for the ADMP510 microphones. Note the flat response over the targeted band 600 2400 Hz

tial sampling through arrays, analogous to the Nyquist frequency for sampling in time8. If the Nyquist criteria is exceeded, then additional beams will appear that are equal in magnitude to the main beam. This is known as spatial aliasing, and an example may be seen in Figure 7.

With this knowledge in mind, we analyzed SDP16s array. The SDP16 team used a nested array as pioneered by Smith9, where the targeted frequency band was split into smaller bands, and then subarrays were constructed out of the 16 available microphones. By sharing microphones between subarrays, each subarray could be allocated 8 microphones.

Band	Highest Wavelength to Mic Distance Ratio
600 - 1000	0.617
1000 - 1700	0.69
1700 - 3500	0.72

TABLE 2
Project Sauron Frequency Bands

However, for the SDP16 array, within each band, approximately half of the frequencies would exceed the Nyquist Criteria, as shown in Table 2:

Figure 9a and 9b demonstrate the negative effects of exceeding the Nyquist criteria. As our system implements beamforming in parallel, the spatial aliasing would add even more noise, as the additional beams will each produce an aliased beam.

To correct this issue, we divided the targeted frequency band into octaves, as Smith originally did. For a frequency band $[f_L, f_H]$:

- For each octave $[f_{iL}, f_{iH}]$, a subarray was created with microphone distance di = $1/(2f_{iH})$, to avoid aliasing
- Each successive subarray had a microphone distance d_i = 2d_i-1, to share as many microphones as possible
- All subarrays were allocated the same number of microphones, to ensure that the array response to each band was identical

With these requirements, we could create three arrays targeting [600, 1200].,[1200, 2400], and [2400, 4800] Hz. By eliminating the highest frequency band, we could increase the number of microphones allocated to the lower bands from 8 to 11 microphones. Internal tests among the team found that speech was still intelligible lacking the higher frequencies.

The array performance is summarized in the table below, for the best and worst cases frequencies:

d/λ	Steering Angle	Beamwidth
$\lambda/4$	90	18.5
	150	36.9
$\lambda/2$	90	9.3
	150	18.5

TABLE 3
Earbeamer Array Performance

As shown in Table 3, beamwidth suffers when the steering angle is directed towards its maximum angle of 150^o. However, beamwidth in the regions directly in front of the array, from 60 120 degrees remains relatively close to the specification. This is likely the best performance we can achieve with our current 16 channel Analog to Digital converter. From Equation 4, the only way to narrow beamwidth further is to add more microphones, but that would require purchasing an ADC that is outside the range of our budget.

2.3 Beamforming Algorithm

Our beamforming algorithm uses a delay-sum technique, where a varying delay is applied in software to each microphone before summing each signal together. By choosing the correct delays based on the position of the audio source, we can align the phase of the signals from a certain location causing constructive interference, thus theoretically amplifying only the audio from that location.

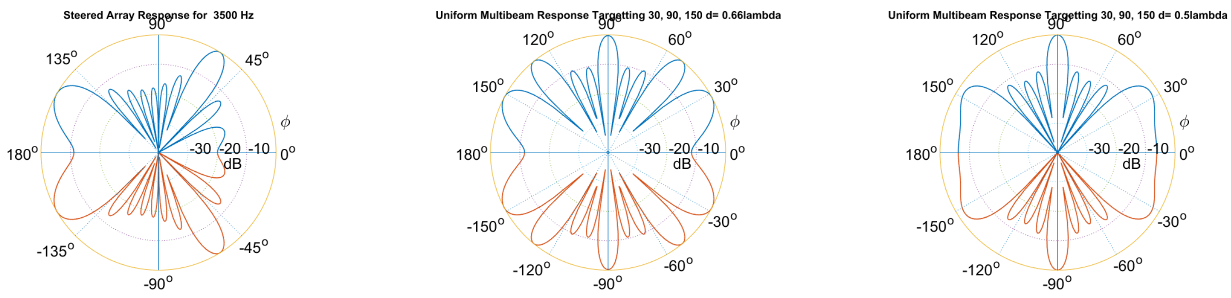
$$y[n] = \Sigma_{m=0}^{M-1} x[n - m\tau] \tag{5}$$

The beamforming algorithm was implemented in C++, using a pipelined approach. The idea behind the pipelined approach is that we can be constantly reading in audio data, which has a I/O cost associated with it, while performing calculations on the previously received audio at the same time. This way, we would be given the sample length to perform all our calculations. Using a sample rate of 16kHz and a sample size of 1024 samples, each buffer would hold 64ms of audio data.

One of our concerns was the amount of time the beamforming calculations would take for a single beam. As we are interested in calculating multiple beams, it is important that we can perform the calculations much faster than the time each sample takes. After running tests on our beamforming algorithm, we calculated that on average it takes under 200μs to perform the beamforming calculations for one beam, well under the 64ms we have.

2.4 Anti-Aliasing Filter and ADC

The purpose of the anti-aliasing filter block is to cutoff sounds after a certain frequency. This filter comes before the



(a) a (b) b (c) c

Fig. 9. The left plot shows the SDP16 arrays response to a 3500Hz signal when the array is steered to 150^o, creating an aliased lobe at approximately 60^o. The center plot shows SDP16s array response when beamforming is performed in parallel, and aimed at 30, 90, and 150 degrees. The spatial aliasing at 135 and 55 degrees adds a significant amount of unwanted amplification to the field. The far right plot shows the highest frequency for the Earbeamer array, pointed at the same locations, with no spatial aliasing

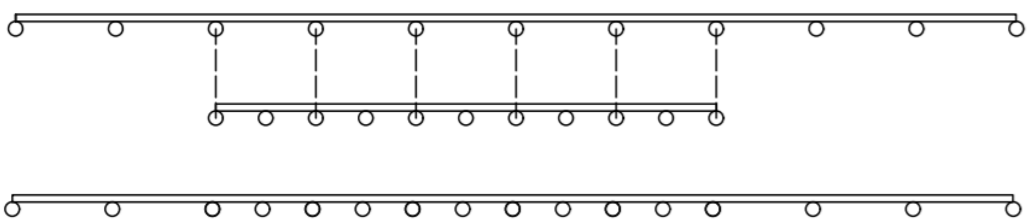


Fig. 10. The Array Geometry for the new Earbeamer array. 16 microphones are shared between a [600,1200] band with d= 7cm, and a [1200,2400] band with d = 14cm

Analog to Digital Converter, a device that converts the signal from analog to digital before being sent to the computer. One requirement for the filter is to have a delay response that does not interfere with the beamforming algorithm. The low pass filter will be built on a circuit board using capacitors and inductors using a technique that is used for microwave frequencies.

Rather than building an RC series filter which gives a 3Db cutoff at the desired cutoff frequency, it is preferred to have a sharper drop off at the cutoff frequency. In order to do that, we can use a method called the Insertion Loss Method. This method is designed to combine capacitors and inductors on a circuit board to achieve a sharper drop off of attenuation. The Butterworth filter, as shown in Figure 9, does not drop off in attenuation until we reach the cutoff frequency. The more capacitors and inductors added together in the circuit, the sharper the drop off, but it comes at the expense of more elements on the circuit, which could lead to more room for error. The issue with achieving a sharper drop off is the greater the group delay. Group delay is the rate of change of transmission phase angle with respect to frequency. The linear phase filter, another circuit using capacitors and inductors, can achieve a very good delay response at the expense of a very slow attenuation drop off at cut off frequency. As mentioned previously, the anti-alias filter should provide a sharp drop-off in order to avoid aliasing. Once the message is sampled, there is no way to undo aliasing. However, we also have to keep in mind the delay response of the filter as well, as it can not interfere with the beamforming algorithm. A suggestion made by Professor Kelly was to design one filter that focuses solely on getting a sharp drop off in attenuation, and another filter followed right after that achieves a good delay response. The cut off frequency we are aiming for is 8 kHz because our sampling frequency is at 16 kHz.

The Analog to Digital Converter comes after the filtering process is done. We use Visual Studios to implement sampling at 2 times our bandwidth, or 16kHz. The signal that was once analog is converted to a digital signal.

2.5 Xbox Kinect

A key component of our hearing aid system is the ability to identify targets within the room, and determine the location of each target relative to an array of microphones. These coordinates are used to dynamically aim our beamforming algorithm as targets move about the room. To accomplish this, we needed a robust computer vision system with a depth sensor that had adequate range to cover a typical living room. The Microsoft Kinect for the Xbox One fulfilled these requirements.

The Kinect uses a Time-Of-Flight system to gauge depth. An infrared emitter emits pulses of infrared light, and an infrared sensor records when the pulse is reflected back. By recording the time required for the reflection to arrive, the relative distance of a point in space may be calculated. Using this system, the Kinect is able to maintain an effective range

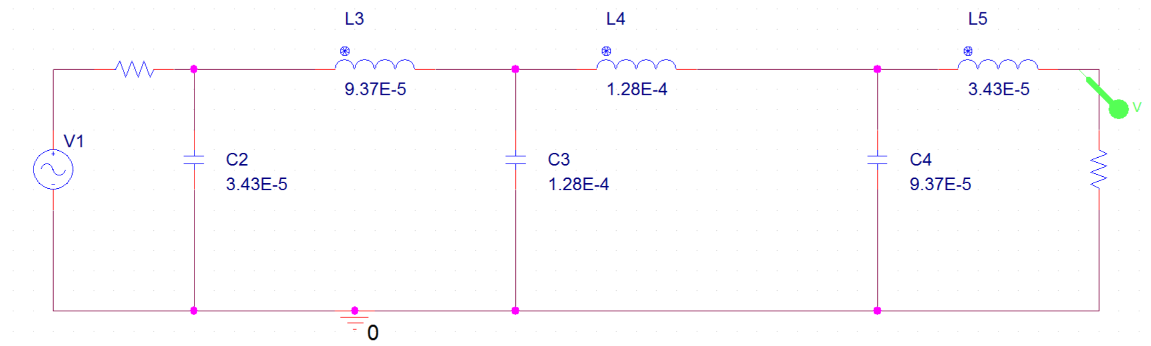


Fig. 11. Butterworth Antialiasing Filter

of 0.5 to 4.5 meters, which more than meets our minimum range specification of 20 feet. Further, the TOF system used in the Kinect for Xbox one has been proven accurate to a depth of 1mm [4], which is more than adequate resolution for aiming the beamformer.

To test the skeleton tracking, a program was written to extract coordinates from any target that entered the field of view, calculate the relative angle of that target to the Kinect, and display the angle on a screen. Through the course of this testing, it was found that the angle of view of the Kinect was only about 60^o, as shown in Figure 9. Originally, we intended to place the Kinect directly below the microphone array, so that the origin of the Kinects coordinate system would align with the center of the array. However, our specification requires that any target from a 30^o to 150^o degree angle be selectable. To accommodate this, we must offset the Kinect to have a better view of the room, and then translate the returned coordinates to a system with the array center at its origin.

2.6 IPhone Application

The phone application is the single point of interaction for the user. It communicates with the central processing software running on the computer. Web socket technology has been chosen to allow for communication between mobile application and computer application. This is a versatile, platform-agnostic protocol that is supported on many platforms. More importantly, it allows two-way communication in an arbitrary manner either party can send a message to the other at any time.

To create the application, an understanding of iOS application development is required, which is something that must be learned. Specifically, development will be done in the Swift programming language using the Xcode IDE. This was chosen as it is modern and well-supported, targeting interactive application development of all kinds. Software development techniques learned in the past have been and will be useful in learning this new platform and problem-solving during the development process.

The computer will run a server to stablish a connection with the application over a network. This web socket server is created running on the node.js JavaScript environment. It listens on a port to establish a link with clients, in this case the application. The socket.io engine was used to implement this functionality. In addition to support for running the server with node.js, there is a socket.io library available for iOS Swift development which was integrated into the mobile application.

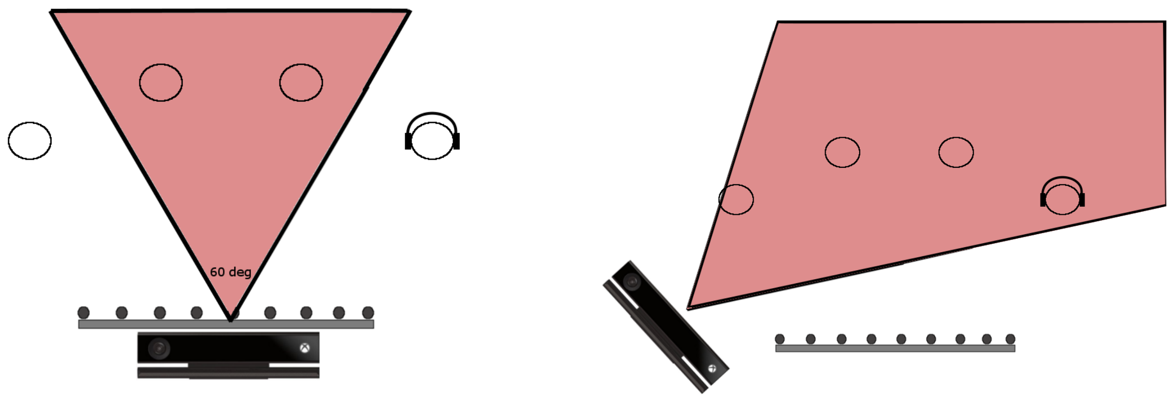
What the user sees is a graphical display of each target as shown in the figure. The room layout on the display will have a fixed orientation, with a fixed reference point indicated. When the target locations are updated on the main processing software, they will be sent by the server to this application. These changes will be reflected by moving the position of each target indicator on the phone. Each target displayed on the application can be pressed, and when it is, the phone sends this information to the server which alerts the system that a change in processing is required either to enable or disable a specific target.

To test the functionality of this subsystem a mobile application was designed and created. Then a server was set up and the application was deployed on an actual iPhone device. Both devices were connected to the same WiFi network. Pressing one of the target indicators on the app generated messages on the computer, reflecting which target was toggled. This showed that arbitrary communication was possible between the two devices. Further work must be done to make the app more interactive and dynamic.

3 Project Management

The Ear Beamer team has successfully met its MDR objectives with deliverables beyond the initial plan shown in Table 4. All subsystems have been shown to work successfully and the next objective is complete system integration. With an initial focus on modelling beam-forming performance, we continued working to implement the hardware subsystems. Currently the software and hardware subsystems are working successfully with synthetic inputs, without communication with other parts of the system directly. They are all reading or processing data appropriately but must be patched together before the system will be fully functional.

Members of the Ear Beamer team have been assigned specific roles as shown in the figure, but continue to assist each other as needed. Weekly team meetings and adviser meetings help facilitate continued progress. In addition, the team communicates frequently using a chat application called Slack, with occasional Skype group video calls.



(a) a

(b) b

Fig. 12. The Angle of View is too narrow to adequately cover a room, so the Kinect must be offset from the microphone array to identify all possible targets

The Ear Beamer is proceeding on schedule. We plan on having a working product and be able to present it for CDR in March. After CDR, we plan to implement Chebychev weighting on microphones and prepare for demo day in May. The full schedule for completing the project is shown in Figure 11 in the Appendix

4 Conclusion

Currently, we have many of the subsystems working on their own, and we must now tie them together. We have planned for this, and it should be relatively easy to accomplish this. We anticipate encountering issues however, as most of our subsystems have been tested with simulated data, and we expect that real-world data might cause variances in our results.

Appendix A

Appendix one text goes here.

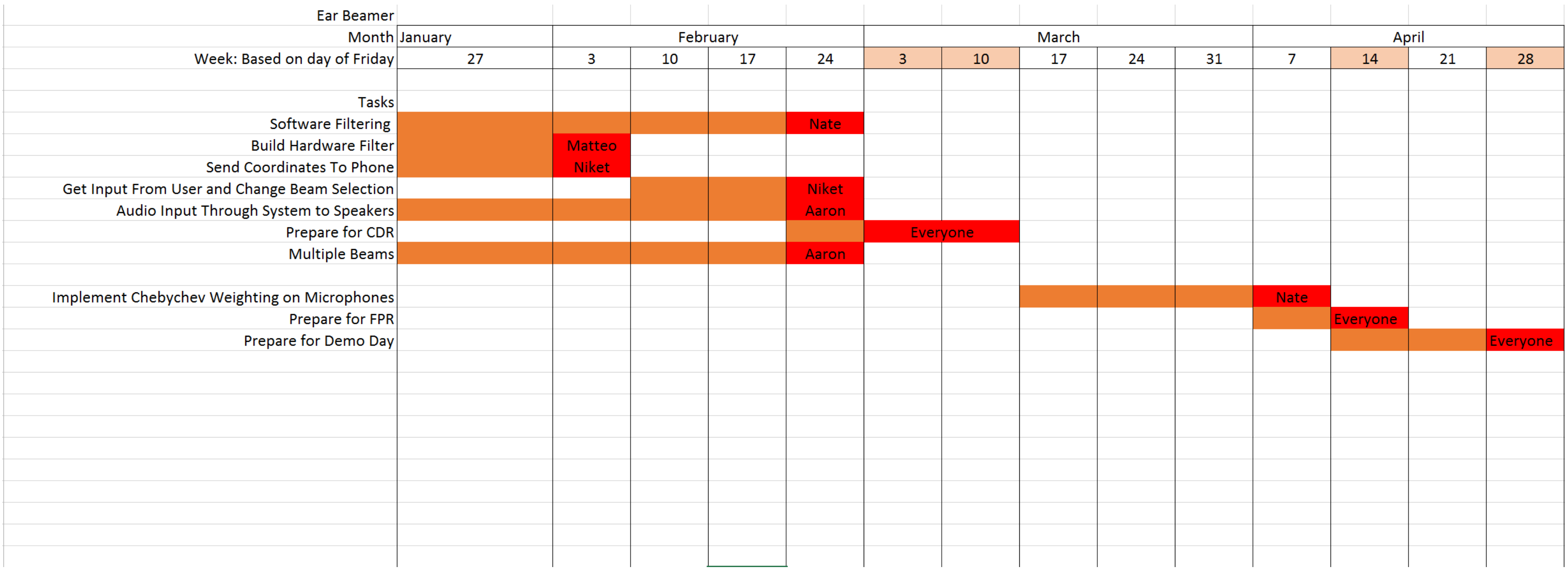


Fig. 13. Gantt Chart for the Remainder of the SDP Projet

APPENDIX B

Appendix two text goes here.

The authors would like to thank...

REFERENCES

[1] S Kockkin. "MarkeTrak V: Why my hearing aids are in the drawer: The Consumer's Perspective". In: *The Hearing Journal* 53 (2000), pp. 34–42.

[2] Jeff Rodman. *The Effect of Bandwidth on Speech Intelligibility*. Tech. rep. Polycom, Inc, Jan. 2003.

[3] FR. Lin et al. "Health ABC Study Group FT. Hearing Loss and Cognitive Decline in Older Adults". In: *JAMA Intern Med* (2013), pp. 293–299.

[4] Cecilia Chen. *Verifications of Specifications and Aptitude for Short-Range Applications of the Kinect v2 Depth Sensor*. NASA Glen Research Center. Slide Deck. Sept. 2014.

[5] Walter Brown et al. *Sauron Security Final Design Review Report*. Senior Design Project. Apr. 2016.

[6] National Institute on Deafness and Other Communication Disorders. *Age-Related Hearing Loss*. Tech. rep. 97-4235. US Department of Health and Human Services, National Institutes of Health, 2016.

[7] David Myers. "Silently Suffering From Hearing Loss Negatively Affects Quality of Life". In: *American Psychological Association* (Oct. 2016).

[8] Sophocles Orphandis. "Electromagnetic Waves". In: 1st ed. Rutgers University, 2016. Chap. Antenna Arrays, pp. 1089–1090.

PLACE PHOTO HERE

Michael Shell Biography text here.

John Doe Biography text here.

Jane Doe Biography text here.