SmartHearing: An Artificial Intelligence Beamforming Hearing Aid System

Suraj Anand

2017-2018

Science Research

Table of Contents

ANAND, Suraj
Computer Science
2018

SmartHearing: An Artificial Intelligence Beamforming Hearing Aid System

Age-related hearing loss is becoming an epidemic with the rise of deafening speakers and headphones. As current hearing aids blend background noise with speech requiring months of adaptation, 80% of adults ages 55-74 years who would benefit usage do not wear aids, often resulting in cognitive decline and a lower quality of life. The goal of this study is to construct a hearing aid system with improved speech quality versus current models. The initial steps were to suppress interference from a location not in front of the wearer and to decrease noise.

Microphone array acoustic beamforming techniques were utilized to suppress sound that did not arrive from directly in front of the hearing aid wearer. In addition, a backend software noise reduction technique was sought to further enhance significant speech. Machine learning procedures including the least mean squares, spectral subtraction, independent component analysis, and a convolutional neural filter (wavenet) were tested to determine the most effectual and inexpensive algorithm with an array of simulations. Finally, the best performing algorithms (beamforming frontend and software backend) were programmed to a Raspberry Pi 3.

The two-microphone broadside array beamformer reduced noise located on the sides of a person by about 3-7 dB depending on the relative angle to the microphones. The wavenet exhibited the most successful results for backend speech enhancement, reducing noise by an additional 8 dB on average. These algorithms produced crisper, clearer speech from the true source and greatly reduced noise interference.

This hearing aid system enhances speech. These algorithms provide a valuable base for a real-time hearing aid which can be utilized by patients dissatisfied with current hearing aids. While in the past, hardware and battery were limited, novel technology such as AI specific chips will be able to inexpensively harness these methods to construct a smarter hearing aid system.

Introduction

Age-related hearing loss has become an extremely potent problem with the rise of media and speakers. The rise of household media and external speakers has drastically increased the necessity for a well-designed hearing aid; presently, the most significant sources of sensorineural hearing loss result from the chronic exposure to concerts, drills, headphones, and other such events and appliances rather than the workplace as in the past. Sensorineural hearing loss results in hearing muffled sounds, blending in with background noise.

Due to a few fundamental flaws in most hearing aids 80% of adults at ages 55-74 years who would benefit from hearing aids do not wear them. A review of hearing aids from 2012 estimates that close to 23 million adults in the United States contain hearing loss (classified as a Pure Tone Audiometry of 25 dB hearing level or greater in both ears) (Mccormack, et al., 2012). Additionally, those fitted to hearing aids often time do not utilize them. Although various reasons exist for this phenomenon including comfort and maintenance, a high-prevalence cause of this is “the hearing aid not providing enough benefit” (Mccormack, et al.). Present compressive amplification hearing aids merely amplify sound, rather than attempting to enhance the sound to differentiate it from background noise.

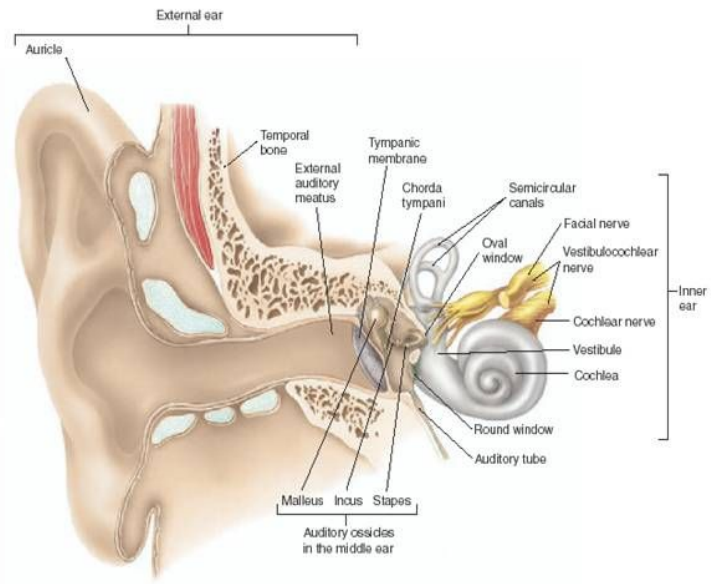


Figure 1: The Anatomy of the Ear (Kujawa, et al.)

Hearing is a complex physiological process in which sound waves are translated into electrical action potentials received by your brain. Sound is captured by the auricle and auditory canal. Inthe middle ear, sound produces vibrations in tympanic membrane, subsequently amplified by the three auditory ossicles. Vibrations to the basilar membrane bend upwards in order to generate action potentials of the cochlear nerve; these action potentials are further analyzed in the temporal lobe of the cerebrum (Physiology). Sensorineural hearing loss (SNHL) is the damage or death of the sensory receptor hair cells in the cochlea and relaying neurons.

Sensorineural hearing loss is multifactorial, with common causes including "overexposure to sound, certain drugs, infection, or immune-induced inflammation" (Wong, et al., 2015). This hearing loss does not affect all frequencies equally, shown by figure two. Higher frequency sounds are greatly diminished, while low frequency sounds remain audible in SNHL. This is the main drawback of hearing aids. Consisting of a microphone and speaker (also background noise cancellation in some cases), hearing aids work to amplify all sound that travels into the auricle. The dilemma is that SNHL causes sounds to appear muffled, blended in with the background noise. When all noise is amplified, background noise and speech are both heightened, causing difficulty in recognizing speech, especially in environments with babble. This indistinctness is more

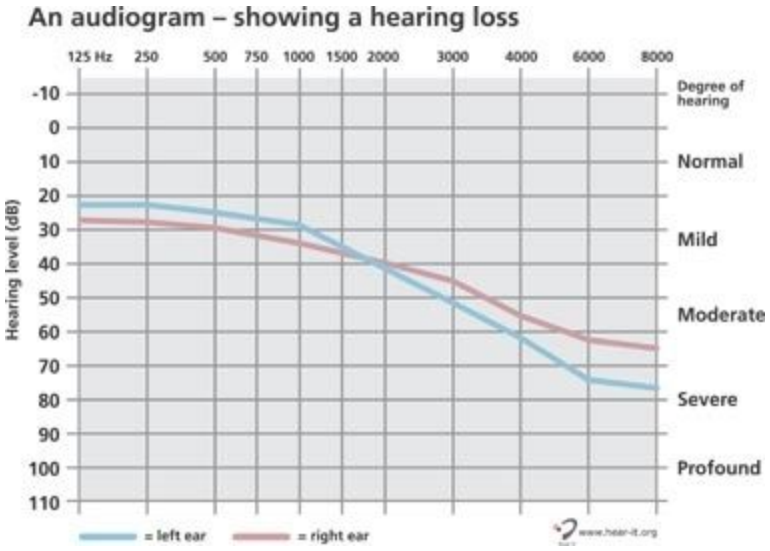


Figure 2: Typical Hearing Loss Audiogram (Hear-It)

prevalent in higher frequency sounds such as s, h, and f. This complication is one of the most significant reasons associated with nonuse of hearing aid technologies.

Microphone array beamforming is a technique useful in suppressing background sound. Beamforming focuses a specific beam on sound from a certain predetermined angle of the microphones. All other sound is attenuated. This technique is useful as a hardware-implemented frontend that results in suppression of the sound extruded from locations other than the predetermined angle.

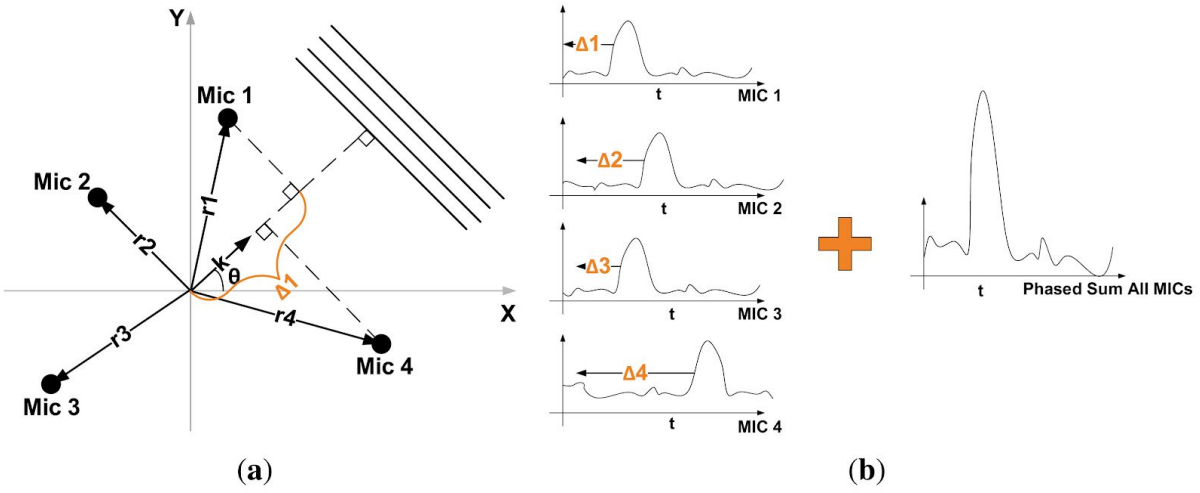


Figure 3: Delay-sum beamforming example (Tiete J, et al., 2014).

Shown in Figure 3, sum and delay beamforming is the technique of calculating delay in the source vectors and then summing them. This procedure works as the when multiple isometric microphones are placed relatively (a few centimeters is sufficient) far apart, the sum of the signals provides a stronger signal gain for sources that are at 0^{o} compared to 90^{o} or 270^{o} (Greensted). Therefore, when the summed signals are combined and the true source is at 0^{o}, the noise and interferers are attenuated. This technique can be applied to this problem to produce

viable results as the user would most likely by facing the person that they are communicating with.

Numerous other methods would also benefit the overall system as a backend to amplify speech. These techniques include recurrent denoising autoencoders (deep and not), spectral subtraction, Independent Component Analysis and other Blind Source Separation methods, and the Least Mean Squares (LMS) algorithm. These methods would be deployed on top of the beamforming technique to provide further attenuation of unwanted noise. Each of them suppresses noise by different means.

Convolutional filters mimic the temporal nature of speech by employment of recurrent connections. If the network is *deep*, complex relationships between clean and noisy utterances are formed by training on a multitude of environments and noise levels. However, deep networks are much more computationally expensive than regular neural networks; in a real-time system, deep neural networks might be too slow, causing in delayed sound from the aid to the person. Delayed sound produces the effect of hearing two separate streams of audio, which is not the desired effect. Thereby, both convolutional denoising filters and deep architectures will be built to optimize the expensiveness and effectiveness.

Convolutional filters aim to learn the rich complexity present in noisy utterances to map a function f(x)→ y with the noisy utterance as x and the clean utterance as y. In the study *A Wavenet for Speech Denoising*, the squared error of the output is minimized. Another similar-use network, a single layer denoising autoencoder employs the the equations $\hat{y} = V h^{(1)}(x) + c$ and $h^{(1)}(x) = \sigma(W^{(1)}x + b^{(1)})$ are utilized where V and W(1) are optimized weight matrices and c and b(1) are optimized bias vectors. The commonly used logistic function $\sigma(z) = \frac{1}{1+e^{z}}$ allows the output y-hat to contain nonlinearities and places each value in h(1)(x) as a value between 0

and 1. This model deploys a small temporal context window to prevent overfitting and to increase computational efficiency.

A recurrent implementation of this would compute representations of the next time step by also employing the last time step multiplied to some weight matrix U, producing the equation $h^{(1)}(x_t) = \sigma(W^{(1)}x_t + b^{(1)} + Uh^{(1)}(x_{t-1}))$ for a single time step. The added element of recurrency "models the temporal dependence which [is] expected to exist in noisy speech utterances" (Maan, et al.). In a multi-layer network as shown in Figure 4, each output of the logistic function would be multiplied by some weight matrix, added a weight bias, and added the scaled previous time step. The resulting matrix would again pass through the logistic function and the process would repeat.

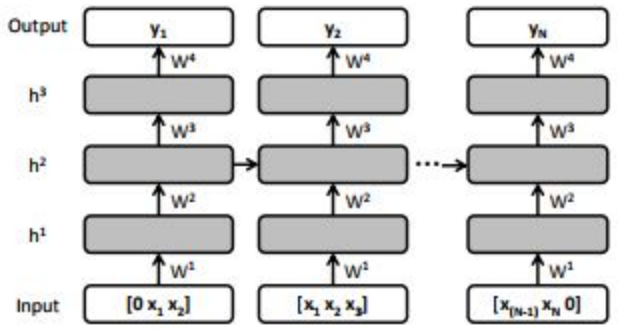


Figure 4: Recurrent Autoencoder Layout (Maan, et al., 2015)

Training convolutional filters on the professional Aurora 2.0 database outperforms the traditional SPLICE algorithm, Wiener Filter and performs well on unseen environments (Maan, et al.).

Additionally, the Least Mean Squares algorithm is another option for a backend. Given some noisy signal and desired signal, this algorithm determines the filter coefficients for a function that produces the least squared error in the audio by stochastic gradient descent.

This adaptive filter works by calculating the partial derivative with respect to the individual entries of the filter coefficient matrix (weights). The assignment function shown by the equation $w := w - \eta \nabla Q_i(w)$ represents each step in descent. As the algorithm continues throughout the training set, the update is computed for each example, the points are reshuffled,

and the the updates computed again until convergence. Each individual entry is adapted for by stochastic gradient descent, taking small steps to convergence (individual points) rather than computing the true gradient for every step. Learning rate and the starting point affect the rate and location of convergence as shown in Figure 5.

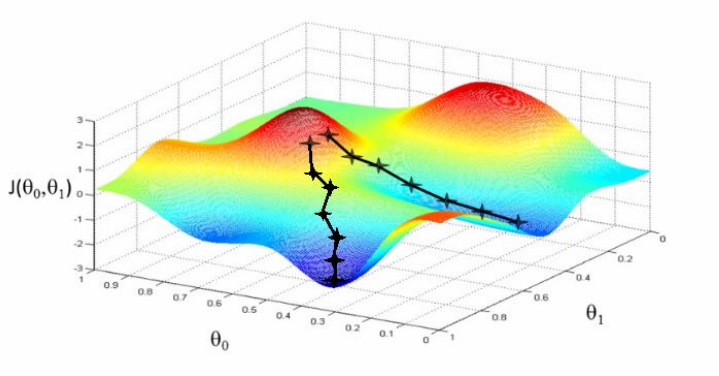


Figure 5: Multi-Dimensional Gradient Descent (Ng, et al., 2014)

Spectral subtraction is yet another viable option for backend attenuation. This algorithm is perhaps closer to the standard, representing the noisy signal as the sum of the desired signal and the noise. Spectral subtraction estimates an average signal spectrum as well as an average noise spectrum. These spectrums are then converted from the time domain the frequency domain with a Fourier transform and to smoothen. The noise spectrum is subtracted from the signal and the signal is restored to the time domain demonstrated by Figure 6.

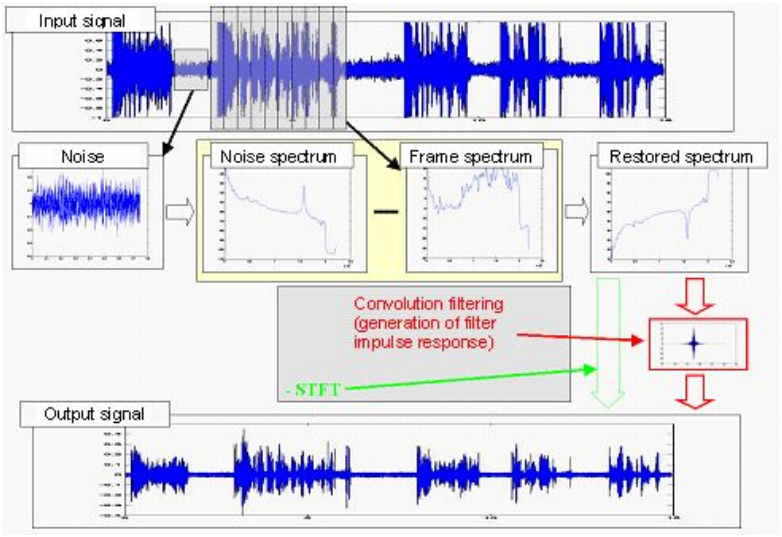


Figure 6: Spectral Subtraction Layout (Gdansk University)

Unfortunately, a downside to this algorithm is that it is “assumed that the signal is distorted by a wide-band, stationary, additive noise”(Multitask Noisy Speech Enhancement

System 2014). The algorithm does not assist in the removal of babble noise, the most problematic and confusing noise for hearing aid users.

Using previous studies, the frontend acoustic microphone array beamforming will be predicted to result in a gain of 6-10 decibels (dB) in noisy environments. A convolutional filter will likely result in the most successful gain, at the expense of the most computations (power-intensive). On the contrary, the Least Mean Squares filter will provide less attenuation than the convolutional filter, yet with less computations. In total, an expected total gain of 12-20 dB can be predicted, performing better than modern hearing aid systems at the expense of between 3-6 milliseconds of delay.

Procedure

Phase I: Software Development

On a local computer or desktop, access to the Aurora 2.0 Sound Database was gained: this was done by first acquiring the TIDIGITS sound database from the Linguistics Data Consortium (https://www.ldc.upenn.edu), and then the Aurora 2.0 Sound Database from the European Language Resources Association (http://www.elra.info/en/projects/archived-projects/aurora/details-aurora-databases/). The makefiles were run and the cfiles compiled on a Linux systems. The scripts to create patterns were run.

MATLAB and python3 were downloaded from the web. Octave may also be used. If so, by the terminal install additional signal, linear-algebra, optim, and struct packages (pkg install –forge package_name).

Microphone Array Beamforming Frontend

A microphone array beamforming simulation was constructed in matlab using different

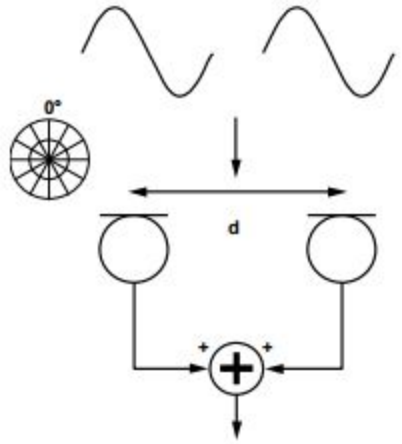


Figure 7: Two Mic Broadside Array
(InvenSense Mic)

viable setups. First, a two-microphone broadside array was modeled.

This setup would utilize two omnidirectional microphones with a distance (d) of 15 cm between them. Sound would be attenuated if it did not come from 0 degrees of the midpoint of the segment connecting the microphones, translating to noise suppression if the sound did not arrive from in front of behind the person. The noise and interference attenuation occurs because when the sound vectors gathered from the two microphones, different delays to each microphone occur: the phases lose congruence, so when summed, the signal strength can be decreased by slight cancellation. The advantages to this procedure are that it is not computationally expensive and relatively simple to implement. The main disadvantage is that the sound from 180 degrees is not attenuated as there is no differentiator between the front and back due to the axisymmetric nature of the array. First, a simple sine signal was determined to model various situations in this scheme and determine the specific amount of noise suppression at multitudinous angles.

Tune different angles of focus and distance apart. The angles from 0 degrees to 360 by increments of 2.5 degree for the main lobe were tested and the dB of suppression were calculated

on and plotted on a polar graph. A logarithmic graph representing dB of attenuation for the angles of 0 degrees, 45 degrees, and 90 degrees was constructed with the model

where f represents the frequency in Hertz, r represents the radius to the sound source in meters, d

$$S(d) = sin2\pi f \sqrt{\frac{rcos(\theta_c)^2 + (rsin(\theta_c) \pm d/2)^2}{c}}$$

represents the distance each microphone is from the midpoint in meters, and c represents the speed of sound in m/s (343 m/s).

Next, a three-microphone broadside array was modeled.

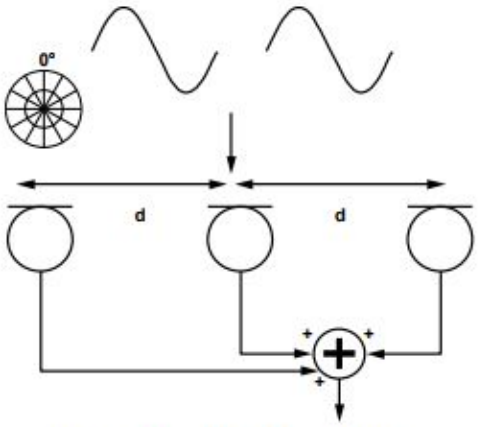


Figure 8: Three Mic Broadside Array (InvenSense Mic)

This setup would utilize three omnidirectional microphones with a distance of 15 cm between them. This setup faces the same disadvantages and advantages as the two microphone model although with greater attenuation. Again, a simple sine signal was determined to model various situations in this scheme and determine the specific amount of noise suppression at multitudinous angles.

Tune different angles of focus and distance apart. The angles from 0 degrees to 360 by increments of 2.5 degree for the main lobe were tested and the dB of suppression were calculated

$$S(d) = sin2\pi f \sqrt{\frac{rcos(\theta_c)^2 + (rsin(\theta_c) + d * n)^2}{c}}, n \in [-1, 0, 1]$$

on and plotted on a polar graph. A logarithmic graph representing dB of attenuation for the angles of 0 degrees, 45 degrees, and 90 degrees was constructed by the model

where f represents the frequency in Hertz, r represents the radius to the sound source in meters, d represents the distance each microphone is from the center microphone, and c represents the speed of sound in m/s (343 m/s).

Finally, a two-microphone endfire array was modeled.

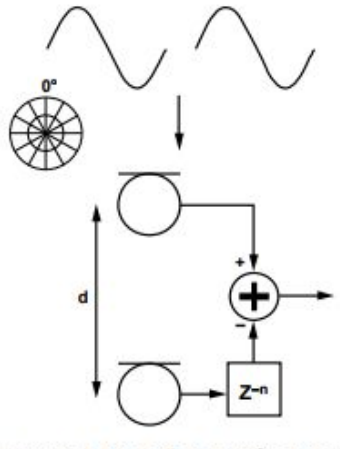


Figure 8: Two Mic Endfire Array
(InvenSense Mic)

This setup would utilize two omnidirectional microphones with a distance of 15 cm between them. This differential array allows for much greater attenuation of rear elements as the ideal sound is located on the same line of field as the microphones. When the delay is properly calculated and signals summed, a cardioid signal should be formed with almost complete suppression of sound behind. Unfortunately, as it would not be practical to wear a microphone

on the front and back of one's head, this setup could not benefit this particular situation. Still, a simple sine signal was determined to model various situations in this scheme and determine the specific amount of noise suppression at multitudinous angles.

Tune different angles of focus and distance apart. The angles from 0 degrees to 360 by increments of 2.5 degree for the main lobe were tested and the dB of suppression were calculated on and plotted on a polar graph. A logarithmic graph representing dB of attenuation for the angles of 0 degrees, 45 degrees, and 90 degrees was constructed by the model

$$S(d) = sin2\pi f \sqrt{\frac{(rcos(\theta_c) \pm d/2)^2 + (rsin(\theta_c))^2}{c}}$$

where f represents the frequency in Hertz, r represents the radius to the sound source in meters, d represents the distance each microphone is from the center microphone, and c represents the speed of sound in m/s (343 m/s).

In order to accurately predict the effect of angle on attenuation, another simulation for the two-microphone broadside array was calculated. A polar plot of the effect of theta on dB level was made by utilizing the same model as the logarithmic plot and calculating the normalized suppression for multiple angles, varying the angle rather than the frequency. This was repeated with the frequencies of 500 Hz, 1 kHz, 2 kHz, and 4 kHz as most speech is below 6 kHz.

To verify the results simulated, two Blue omnidirectional snowball microphones were placed on a line 15 cm apart. A source source was projected 5 feet from the microphones at the angles of 90 degrees, 45 degrees, 0 degrees, -45 degrees, and -90 degrees. The summed signal of the microphone was computed and the true signal magnitude calculated. This verification was repeated 3 times.

Backend

A software backend to accompany the frontend acoustic beamforming is extremely helpful in noise attenuation. This research tested the effects of spectral subtraction, least mean squares, and a convolutional filter to clean the noisy signals.

Blind Source Separation/Spectral Subtraction

Artificial intelligence techniques for blind source separation were attempted. First, unsupervised independent component analysis (ICA) was employed to test for success in source separation. This algorithm attempts to separate two linearly added sources with data from two microphones. Each microphone would receive a different magnitude of each source. From this point, the independent component algorithm employs the two sources of non-random information to blindly and accurately separate the sources. This technique was endeavored three times with the collected data at 0 degrees and 45 degrees from microphone one. The signals were linearly added in this scenario and independent source separation attempted. The resulting signal to interference ratios (SIR) were calculated and compared with the initial signal to interference ratio. Additionally, this technique was tested three times with the collected data at 0 degrees and 45 degrees from microphones' one and two. The signals were used as collected and independent source separation attempted. The resulting SIRs were calculated and compared with the initial signal to interference ratio.

A further algorithm called the DUET algorithm was tried to counter some of the hindrances of ICA. This machine learning method is more commonly found in practice, as it estimates the relative location of the sources and harnesses the same general structure as independent component analysis. From the MATLAB exchange, code for the DUET algorithm was modified to work with custom data. This method was then tried with the same three samples

of linearly added data as the ICA and raw microphone files. The SIRs were collected before and after the algorithm.

Finally, the spectral subtraction algorithm was implemented in Matlab. The background noise was approximated by a moving averages of values taken when no speech occurred (estimated by periods with only mild fluctuations) and then converted to the frequency domain and smoothened by a Fast Fourier Transform, subtracted from the recorded signal's Fourier Transform, and converted back to the time domain. This produced the approximated desired signal. The SIRs were collected from before and after the algorithm and compared.

Least Mean Squares

In MATLAB, the Least Mean Squares algorithm was written by randomly assigning doubles between 0 and 1 to the weight matrices and bias vectors, computing the predicted signal, finding the average squared error, and then subtracting the partial derivative of each individual data point scaled by the learning rate from the weight matrix, so to optimize the weights for

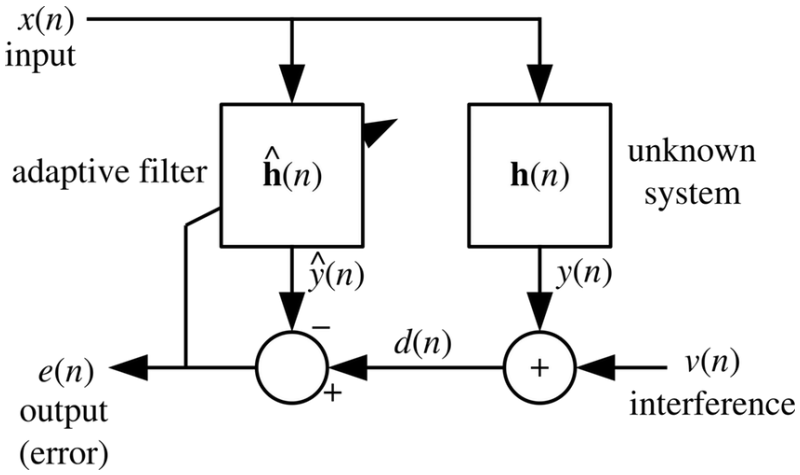


Figure 8: Design of the x-LMS algorithm (Hayes, et al.).

Parameters: $p =$ filter order

$\mu =$ step size

Initialization: $\hat{\mathbf{h}}(0) = \text{zeros}(p)$

Computation: For $n = 0, 1, 2, \ldots$

$$\mathbf{x}(n) = [x(n), x(n-1), \ldots, x(n-p+1)]^T$$

$$e(n) = d(n) - \hat{\mathbf{h}}^H(n)\mathbf{x}(n)$$

$$\hat{\mathbf{h}}(n+1) = \hat{\mathbf{h}}(n) + \frac{\mu\, e^*(n)\mathbf{x}(n)}{\mathbf{x}^H(n)\mathbf{x}(n)}$$

Figure 9: Pseudocode of the x-LMS algorithm (Hayes, et al.).

each individual audio recording. The algorithm was implemented for gradient descent on the previously downloaded AURORA dataset.

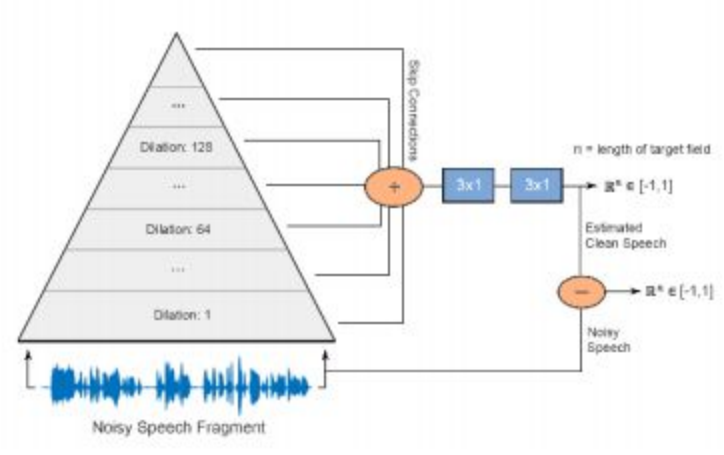
The number of computations and the mean squared error and sum squared error were recorded for three trials of the learning rates, alpha, of .003, .03, .09, .27, .81. Line graphs of the error across each noisy sample were modeled for each learning rate, and the gainage in the signal to noise ratio calculated.

Convolutional Filter

A convolutional filter was constructed from the wavenet library for speech denoising (https://github.com/drethage/speech-denoising-wavenet). References to the folders were parameterized and hyper parameters were tuned. This network was trained on the "Noisy speech database for training speech enhancement algorithms and TTS models" (NSDTSEA) provided by the University of Edinburgh, School of Informatics, Centre for Speech Technology Research (CSTR). The wavenet was a discriminative convolutional neural network offshoot from google's wavenet, utilizing intermediate models to estimate noise rather than to produce speech. A unique energy conserving loss function was used:

$$\mathcal{L}(\hat{s}_t) = |s_t - \hat{s}_t| + |b_t - \hat{b}_t|$$

This allows this algorithm faster and better training times and implementation. Additionally, it is more suited toward source separation that usual least mean squares



techniques. The network iteratively denoised each noisy speech fragment in batches, appending each to the last. This method allows the algorithm real-time applicability.

Figure 10: Layout of the Wavenet (Rethage, et al.)

Sigmoidal gates control activation functions in each layer similar to an LSTM. This captures the temporal aspect of hearing. Furthermore, skip connections (skipping layers randomly) and context stacks were used that enlarge the network without increasing field length (as much as dilation) were employed. The network with dilation factors of 1, 3, and 5 were trained. The dilation of 3 provide the best medium for noise reduction at an efficient speed.

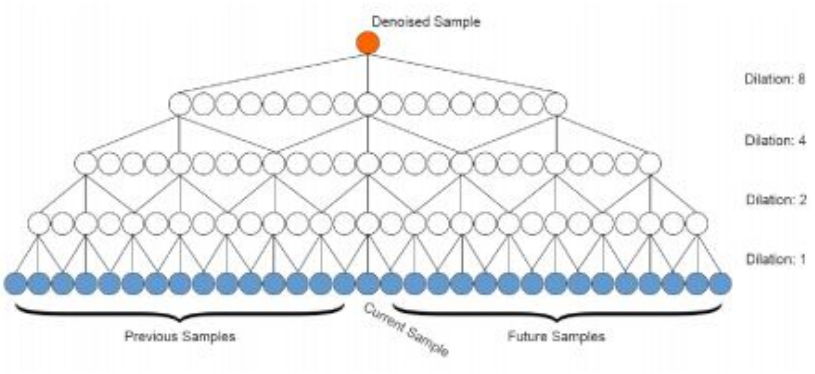


Figure 11: Dilations of the Wavenet (Rethage, et al.)

While the original network added each source sample at the end of the filter as seen in Figure 11, real time application will travel up the net until a smaller dilation so that the sample is split into much smaller batches processed at a time, yet not so small that the output sounds fuzzier than the input. The test lengths of 200, 800, 1600, and 3200 were tried, eventually keeping to 1600 to maximize gain without extreme computations.

Phase II: Hardware Implementation

The two microphones and speakers were wired to the Raspberry Pi 3 model b by soldering. Using the beamforming and wavenet of dilation 3, output length 1600 (highest

cleansing algorithm with computational effectiveness), a simulation to implement the best algorithms on a real-time scenario was coded for in MATLAB. The algorithm was programmed in the raspberry pi and implemented real time. The Matlab software was downloaded to this and run continuously after translated to c code by the Matlab coder.

The device, when started, continuously without interruption runs the beamforming and wavenet algorithms, attenuating non-speech sound that is not location directly in front of the wearer. The differences in perceived sound clearage and signal to noise ratio (the output of the algorithm should sound much better) were recorded for custom environments including babble cocktail party, train, and school, and minor fitting adjustments modified to fit a normal person's head.

Results

The beamforming techniques all provided attenuation at the angles of 45, and 90 degrees: the attenuation remained null at 0 degrees and ranged from -101.2 to 0 dB at the 90 degrees for two microphone Broadside Array and from -132.3 to 0 dB at the 90 degrees for the three microphone Broadside Array. The endfire array, however, provided gainage from the perpendicular angle, and attenuation for angles further than 180 degrees. In addition, Figure 14 demonstrates that higher frequencies experience higher attenuation at angles further from the poles, beneficial considering speech lies at the lower end of the natural hearing range. Furthermore, when this simulation was tested in real-time by the employment of two identical Blueball omnidirectional microphones, the highest gain was located at the 0 degrees with an average of 5.97 decibels gained whereas angles located not at the zenith were effectively attenuated by 3-6 dB; these results are consistent with the simulation.

Figure 15 maps a polar plot of the two microphone broadside as the frequency of the signal rises. This plot exhibits that the magnitude of lateral attenuation increases not only by altering the angle of the source, but also with the frequency of the noise. The frequencies of 500 and 1000 Hz provide little attenuation while the frequency of 4000 Hz provides significant attenuation on the lateral areas.

The backend software for further noise suppression supplied considerable speech enhancement. Supervised learning techniques proved most beneficial for inexpensive, accurate nosie reduction.The Least Mean Squares algorithm, when tested with various stepsizes (alpha), allowed for the minimum Mean Squared Error at the stepsize of 0.81 and showed a general trend

of increased accuracy with increased stepsize (Figure 17). Furthermore, this step size resulted in an average SNR of 7.5609, a typical gain of 4.6206 dB from the average SNR of 2.9403 with no algorithm on the Aurora 2 database. The wavenet convolutional neural filter presented the greatest improvement in noise reduction, as shown in Table 3. The 3 dilation filter typically provided 8.8439 dB of SNR improvement, which translates to the power of the signal improved by a factor of 8 to the power of the sound. Higher dilations attenuated noise and interference superiorly, but at high computational expenses as shown in Figure 18. The 3 dilation filter will be utilized as this network provides a reasonable time delay (about 3-4 ms) with powerful speech enhancements capabilities.

Unsupervised source separation techniques presented less successful results. The spectral subtraction algorithm improved the speech quality less substantially than supervised methods, increasing the mean SNR to 3.8925, which is a 0.9522 increase, most apparent in high noise activities. Some distortion was apparent from this algorithm in low noise, refractive environments. The ICA and DUET algorithms both allowed for almost perfect source separation when the noise was linearly added. Unfortunately, these techniques were not viable in a real-time situation where there is delay between microphones and the SNR of the outputted sound actually decreased with a large amount of harmonic distortion. This is shown in Figure 19 as when a single tone is produced there is accurate representation of the phase delay. However, when speech or complex sounds in a reflective environment are estimated, the accuracy of phase delay prediction significantly decreases, to the point where the algorithm in fact worsens speech intelligibility.

This data confirms that the more computationally expensive methods such as the wavenet often provide the best results. Yet, the lag time is still relatively small allowing hardware technology to be able to implement these algorithms real time.

Harmonic distortion was most significantly heard in the spectral subtraction algorithm, with a few samples of distorted speech. The LMS and wavenet algorithms rarely distorted speech, and only to a mild, almost unnoticeable degree.

Overall, the broadside two-microphone array and the 3 dilation wavenet seemed the most pragmatic to implement, with incredible noise reduction and relative computationally inexpensiveness. These algorithms were successful when programmed to a Raspberry Pi 3 B as shown in Figure 20.

Discussion

The results displayed that the convolutional filter did in fact provide the most gain in addition to being the most computationally expensive. The spectral subtraction performed will in environments with stagnant noise, but distorted silence. The Least Mean Squares did not distort sound, but provided less speech enhancement than the wavenet. A total of about 11-15 dB SNR was gained with the combination of beamforming and wavenet. The initial objective of speech enhancement was successful.

A closer analysis of the trialed algorithms explains the obtained results and algorithm selections. The two microphone Broadside array was chosen for the beamforming solution as it would be too power intensive to implement a three microphone Broadside array and the Endfire array design did not fit scenario. In addition, the temporal process is too complex to viably model by unsupervised machine learning, accounting for the lack of success in the ICA and Duet methods. The highest-performing algorithms were supervised learning techniques as they could capture the nature of speech from previously trained examples as well as adapt to unfamiliar situations. As hypothesized, the convolutional filter enhanced the speech the most. While in the past technology and battery significantly limited the capabilities of high-performing active noise cancellation algorithms, the rise in improved hardware and battery allow for the algorithms tested to be feasibly implemented. Novel neuromorphic chips and accelerated Field Programmable Gate Arrays expedite the process of hardware development for the convolutional

network of the hearing aid system. Beamforming communication between the two hearing aids will utilize similar technology to current Bluetooth between hearing aids and an external phone.

A recent study trained a deep neural network on 10,000 noises to determine the ideal time-frequency ratio mask and then separate sentences from other noises (Cafeteria and Babble) in novel acoustic environments. This research shows promising results for generalization of supervised learning and possible employment in hearing aids, although the algorithm's current hardware implementation is infeasible. Figure 18 shows how precise the DNN trained was in determining the IRM of the signal.

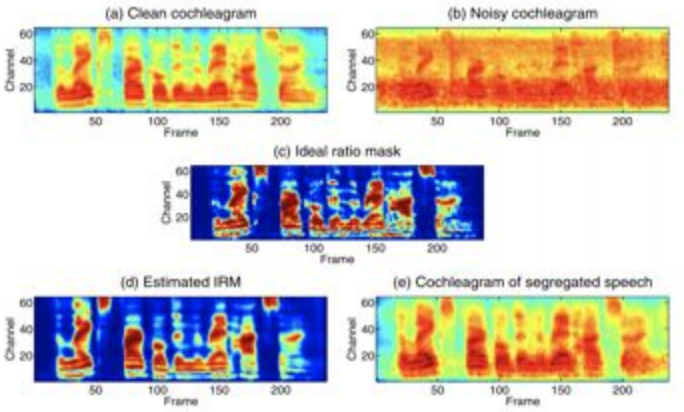


Figure 21: Cochleagram of IRM and separated speech in related study (Chen et al., 2016)

This research confirms the choice of the convolutional filter as an extremely viable solution to speech enhancement, as it possesses the ability to generalize well to novel acoustic environments.

Additionally, another study tested the use of supervised machine learning speech intelligibility increasing techniques perceived benefit on the hearing impaired, providing a 49 point increase in speech intelligibility (could understand 49% more of garbled speech) (Wang, 2015). This study is relevant because although normal hearers can hear the difference in perceived speech, the algorithms are specifically helpful for the speech impaired whose hearing of high frequency speech sounds have decreased and low frequency sounds remained stagnant.

There were a few sources of possible error in this research. The microphones may not have been exactly identical, resulting in imprecise delay estimations when calculated

beamforming and worse results than in reality. In addition, the slight chance of human error may have played a role in the real-time beamforming simulation, as the microphone may have been slightly tilted or some other small nuance. The inherent error of the simulations is always present, providing only a strong estimate of real-time capabilities and errors. Furthermore, the wavenet could have been trained on a more suitable sound database, as well as with more samples.

This design provides the strong advantages of being an extremely useful hearing aid that renders speech crisp and clear. This does not allow background noise to appear blended with speech, which in turn, alleviates the trouble of a few months of adaptation to the hearing aid. By this method, the hearing aid shall become more accessible to the vast majority (with little time to get used to the contraption) and may improve many people's quality of life. In comparison with modern hearing aids, the algorithms are far superior in noise reduction, with the benefit of being much less computationally expensive than even the standard Wiener Filter.

Determination of the best-suited algorithms to perform speech enhancement was a long and intensive process, searching through numerous techniques, most with unviable results. Furthermore, although the design can be utilized in active noise cancellation tasks, this process consumes an unreasonable amount of electricity which would either result in a bigger battery (larger contraption) or a smaller battery life, both major issues with mass production. The algorithms are suited to enhance speech only, which may lead to missing some important event that did not contain speech (most other sound is only suppressed, not alleviated). The same is true of the beamforming functionality: if some important sound event transpires not in front of the wearer, that event will be attenuated. A possible solution to the last two impediments is a

switch that activates the algorithm, which would not be employed during times of jogging, driving, and other such occasions.

Further studies may be conducted to fine-tune the capabilities of this neural network and seek more complex beamforming solutions. In addition, a hardware prototype of these algorithms on a field programmable gate array (FPGA) or other specialized convolutional hardware device (rather than Raspberry Pi) would be beneficial to conduct to create a working product. Testing should be conducted to optimize the Bluetooth for the array beamforming. Finally, the device should be able to turn on and off the speech enhancing wavenet capabilities as they would impede driving. Once a suitable system in built, a matched-pairs black randomized design study should be conducted to examine the device's perceived speech enhancements in comparison with standard systems (no speech enhancement).

This research study is only the first step to generating a more effectual, accessible hearing aid system. The hardware and assembly development of this endeavor still require a massive amount of work.

Acknowledgements

Firstly, I would like to acknowledge Ms. Klose, for aiding in the editing process of the report and board, as well as guidance throughout the year. In addition, I would like to acknowledge my mother, Seema Anand, for helping to understand the research terminology and editing the board. Another person that contributed to this endeavor was Raymond Chen, who provided useful feedback and direction on the initial ideas. Finally, I would like to acknowledge my grandmother, and all the people who have Sensorineural Hearing Loss for giving me inspiration to conduct this research and tolerating hearing loss with unmeasurable optimism.

References

Audiogram – What is an audiogram and how to read it? (n.d.). Retrieved February 19, 2018, from http://www.hear-it.org/Audiogram-

Chen, J., Wang, Y., Yoho, S. E., Wang, D., & Healy, E. W. (2016). Large-scale training to increase speech intelligibility for hearing-impaired listeners in novel noises. *The Journal of the Acoustical Society of America, 139*(5), 2604-2612. doi:10.1121/1.4948445

Conventional Beamforming Techniques. (n.d.). *Springer Topics in Signal Processing Microphone Array Signal Processing,* 39-65. doi:10.1007/978-3-540-78612-2_3

Greensted, A. (n.d.). The Lab Book Pages. Retrieved February 18, 2018, from http://www.labbookpages.co.uk/audio/beamforming/delaySum.html

Hayes, M. H. (1996). Statistical Digital Signal Processing and Modeling. *Wiley*.

Healy, E. W., Yoho, S. E., Chen, J., Wang, Y., & Wang, D. (2015). An algorithm to increase speech intelligibility for hearing-impaired listeners in novel segments of the same noise type. *The Journal of the Acoustical Society of America, 138*(3), 1660-1669. doi:10.1121/1.4929493

Hearing Loss in Children. (2015, July 24). Retrieved February 19, 2018, from http://www.cdc.gov/ncbddd/hearingloss/sound.html

Hybrid Cochlear Implants Aid Specific Hearing Loss. (2015). *ASHA Leader, 20*(11), 12. doi:10.1044/leader.rib2.20112015.12

Maan, A., & Ng, A. (n.d.). Recurrent Neural Networks for Noise Reduction in Robust ASR. *Berkeley*.

Mendel, L. L. (n.d.). Subjective and Objective Measures of Hearing Aid Outcome Lisa Lucks Mendel. Retrieved February 19, 2018, from http://www.audiologyonline.com/articles/subjective-and-objective-measures-hearing-891

M. (2013, December 13). Microphone Array Beamforming. Retrieved September 21, 2017, from https://www.invensense.com/wp-content/uploads/2015/02/Microphone-Array-Beamforming.pdf

Moser, T. (2010). Faculty of 1000 evaluation for Adding insult to injury: cochlear nerve degeneration after "temporary" noise-induced hearing loss. *F1000 - Post-publication peer review of the biomedical literature*. doi:10.3410/f.1551956.1042054

Noise reduction. (n.d.). Retrieved February 19, 2018, from sound.eti.pg.gda.pl/denoise/noise.html

Physiology of Hearing. (n.d.). Retrieved from intranet.tdmu.edu.ua/data/kafedra/internal/normal_phiz/classes_stud/en/stomat

Rethage, D., Pons, J., & Serra, X. (2017). A Wavenet for Speech Denoising. *Cornell*. Retrieved Oct. & nov., 2017.

Wang, Y., Chen, J., & Wang, D. (2015). Deep neural network based supervised speech segregation generalizes to novel noises through largescale training. *Technical Report OSU-CISRC-3/15-TR02*.

Wong, A. C., & Ryan, A. F. (2015). Mechanisms of sensorineural cell damage, death and survival in the cochlea. *Frontiers in Aging Neuroscience, 7*. doi:10.3389/fnagi.2015.00058

Yoon, B., Tashev, I., & Acero, A. (2007). Robust Adaptive Beamforming Algorithm using Instantaneous Direction of Arrival with Enhanced Noise Suppression Capability. *2007 IEEE*

International Conference on Acoustics, Speech and Signal Processing - ICASSP 07.

doi:10.1109/icassp.2007.366634

Appendix

Delay-Sum Beamforming

Table 1: The magnitude gain of the Two Microphone Broadside Array when a 10 dB speech was played at different angles.

Angle (degrees)	Trial 1 Magnitude Gain (dB)	Trial 2 Magnitude Gain (dB)	Trial 3 Magnitude Gain (dB)
-90	3.78	4.61	4.12
-45	3.79	4.64	4.02
0	5.26	6.93	5.91
45	3.75	4.54	4.53
90	3.15	4.41	3.99

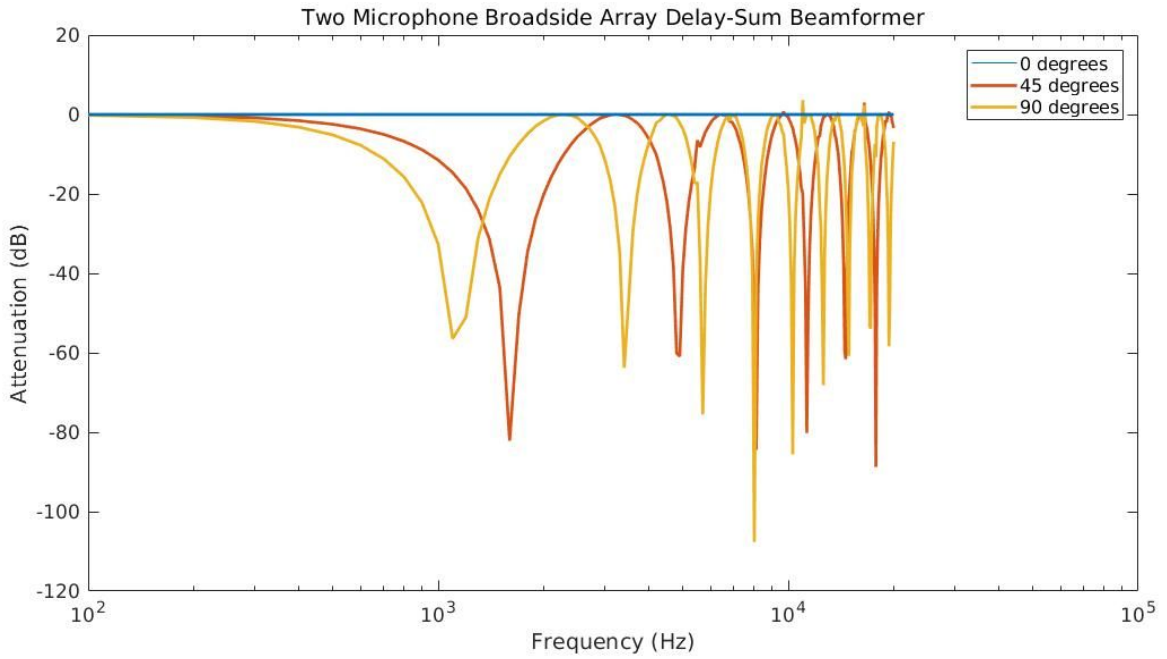


Figure 12: The result of frequency on attenuation of angled noise in a two microphone broadside array delay-sum beamformer

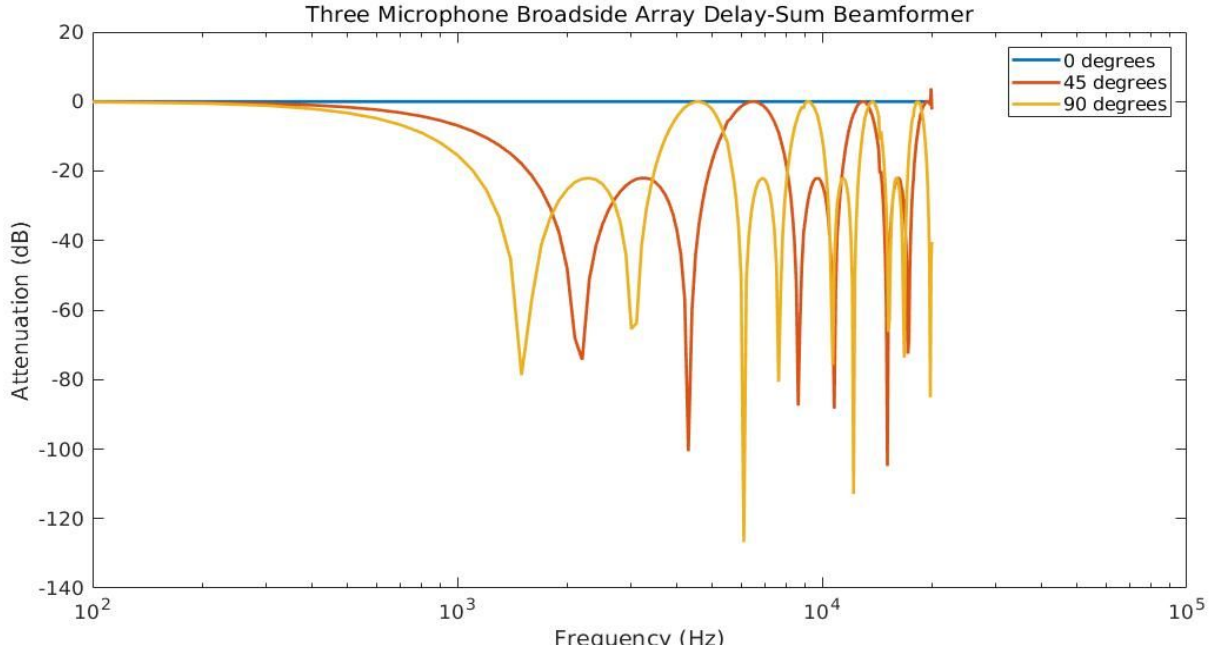


Figure 13: The result of frequency on attenuation of angled noise in a three microphone broadside array delay-sum beamformer

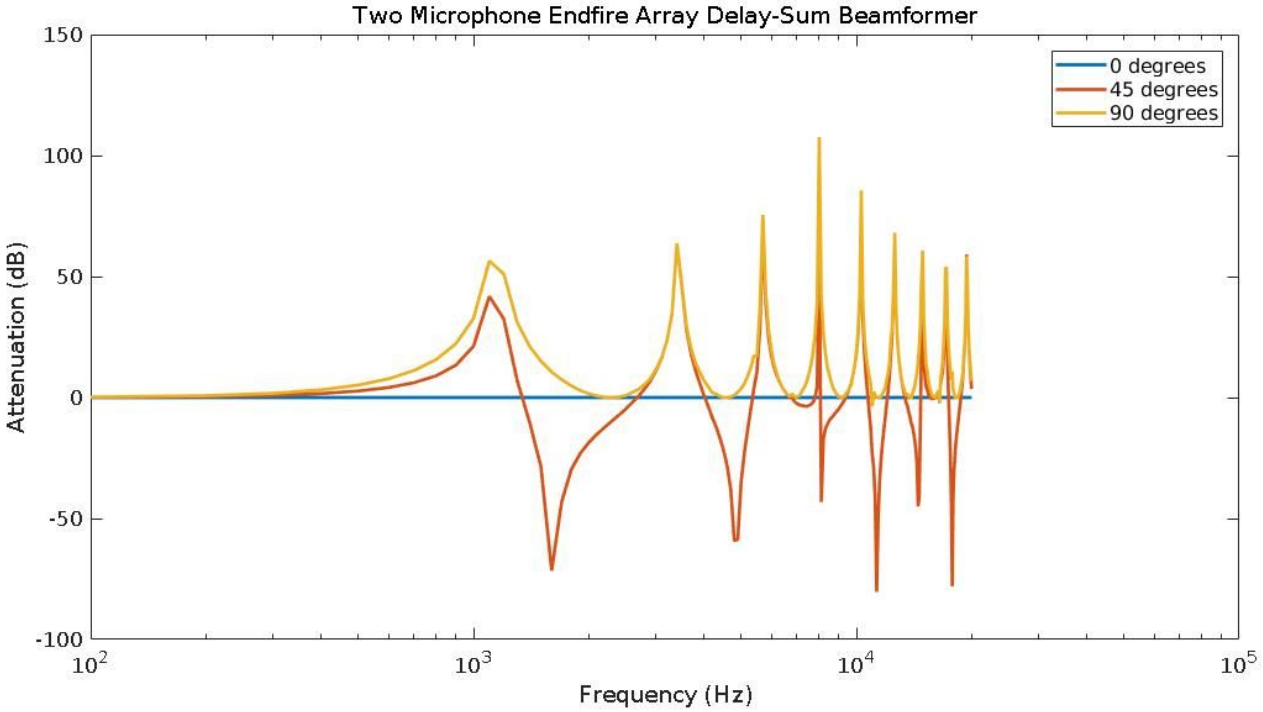


Figure 14: The result of frequency on attenuation of angled noise in a two microphone endfire array delay-sum beamformer

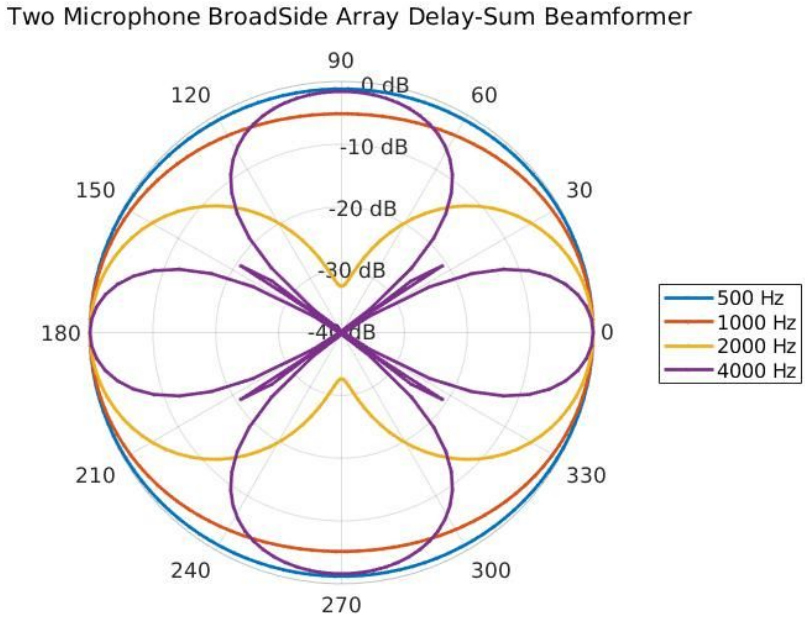


Figure 15: The result of frequency on attenuation of angled noise in a two microphone array delay-sum beamformer

Least Mean Squares

Table 2: The results of the adaptive Least Mean Squares algorithm when applied to the Aurora II Database at various stepsizes.

Stepsize	Mean Error	Summed Error
No Algorithm	4.7187×10^{-5}	11.9600
.003	6.7431×10^{-5}	17.0911
.03	3.7011×10^{-5}	9.3807
.09	2.3490×10^{-5}	5.9538
.27	1.6262×10^{-5}	4.1219
.81	1.4505×10^{-5}	3.6765

*All results generated from the Aurora II Database hand formatted into .wav files with a sampling rate of 8000 Hz, 16 bits, signed-integer encoding, and big endian storage by utilization of the sox sound library. .

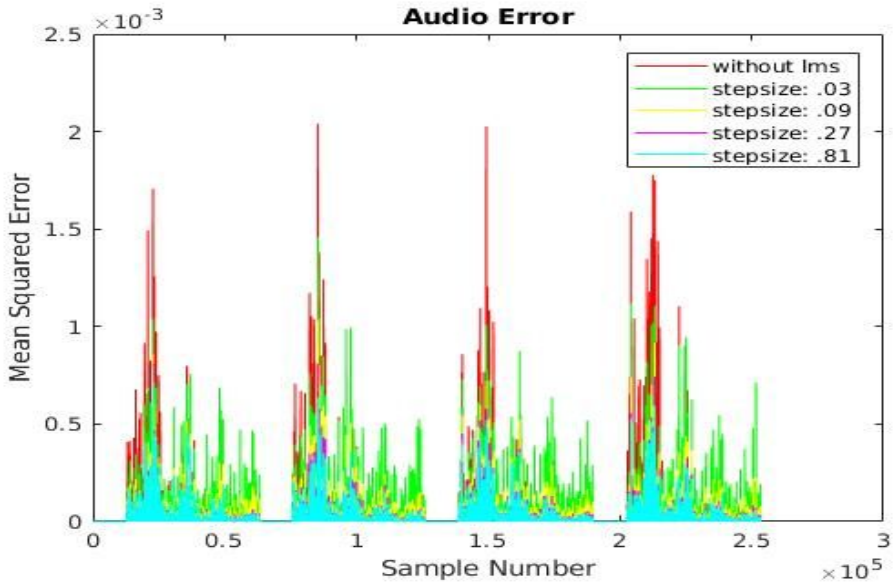


Figure 16: The x-LMS algorithm noise reduction across Aurora II samples.

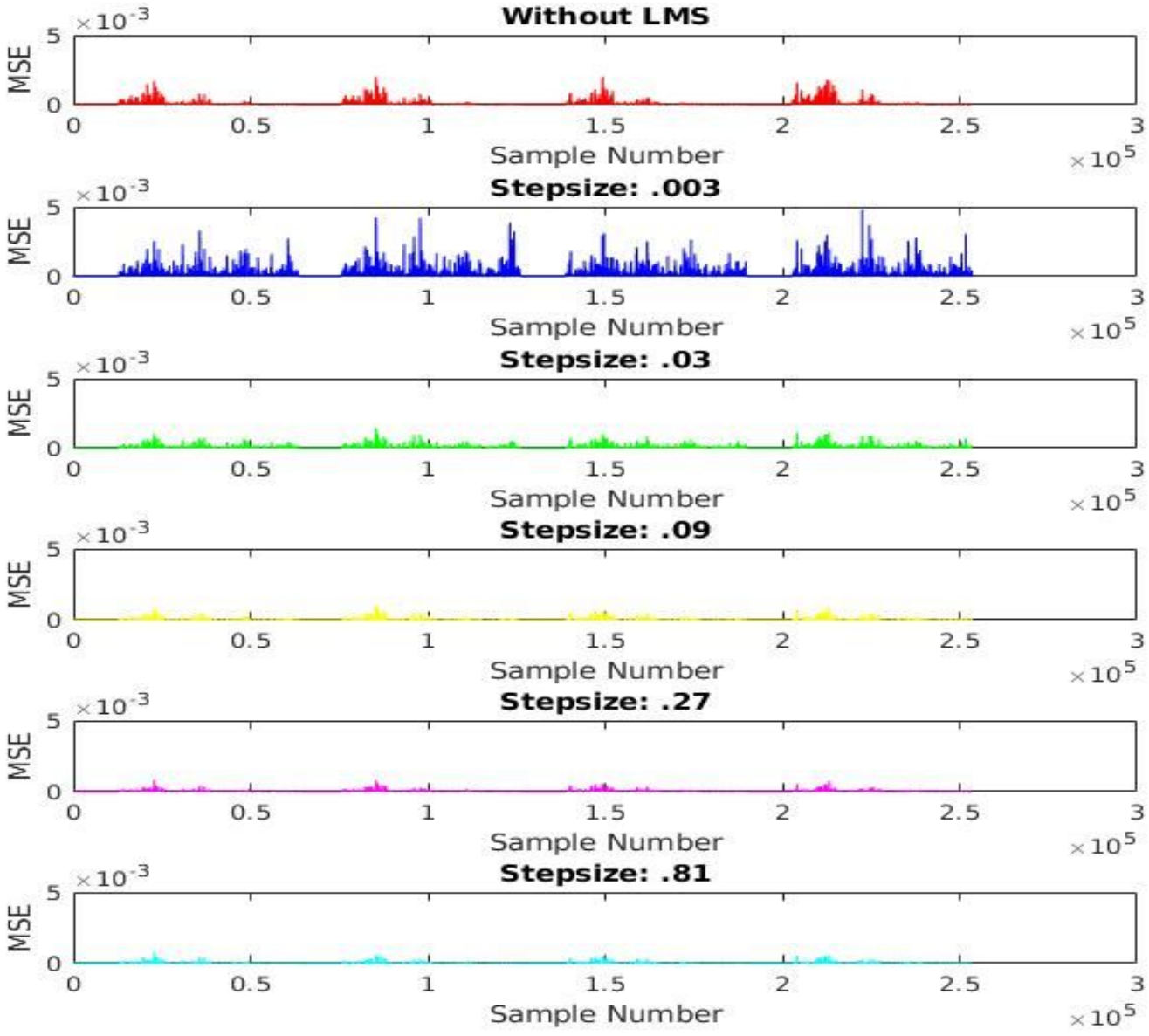


Figure 17: The Mean Squared Error of the x-LMS algorithm for varied stepsizes

Backend

Table 3: The effect of each backend algorithm on the mean SNR of the Aurora II Database.

Backend Algorithm	Mean SNR(dB)
No Algorithm (Control)	2.9403
x-LMS	7.5609
Wavenet	11.7842
Spectral Subtraction	3.8925
ICA	1.293
DUET	3.1023

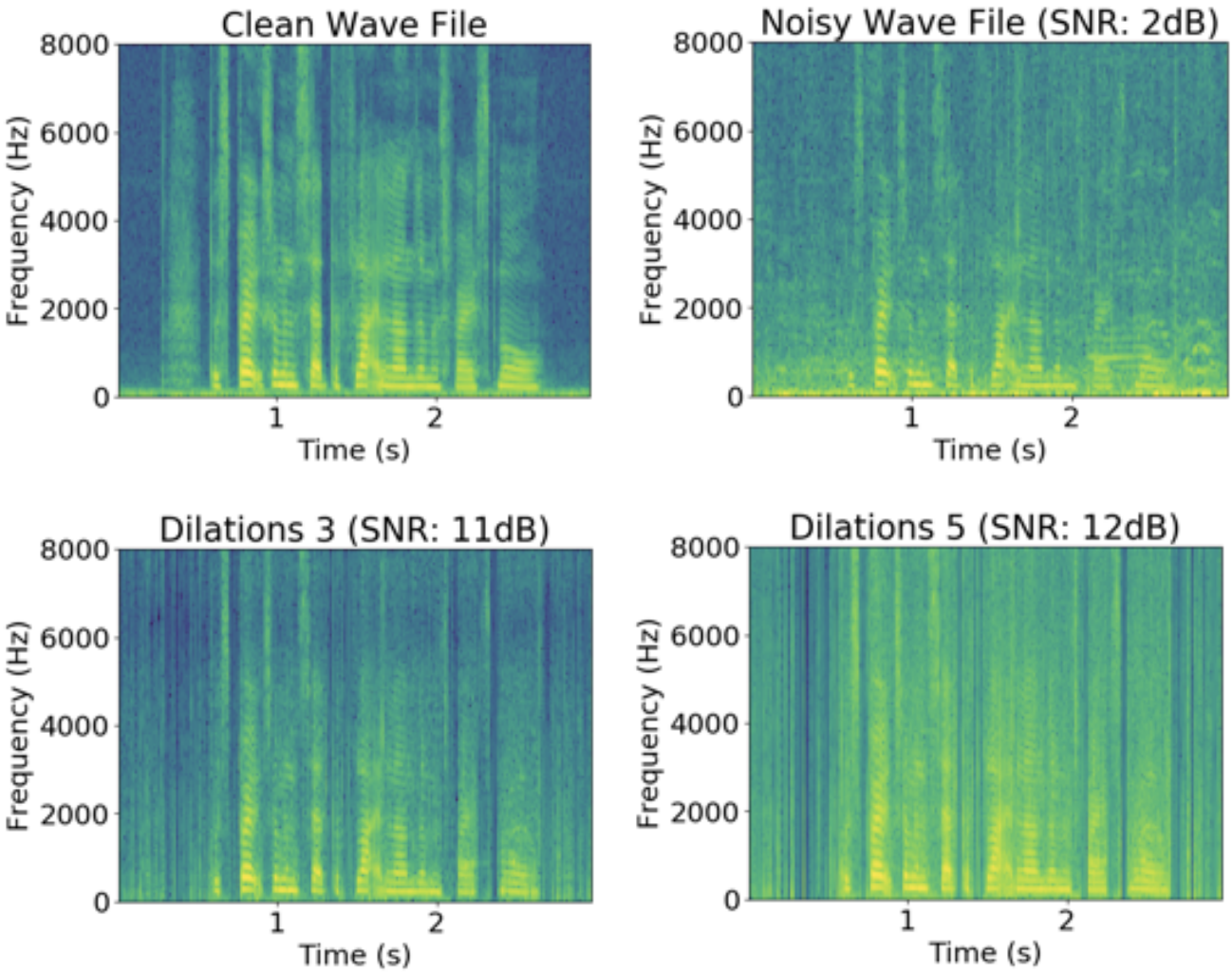


Figure 18: Power Spectrogram Density of Sample p323_023 with and without wavenet processing (dilations 3 and 5)

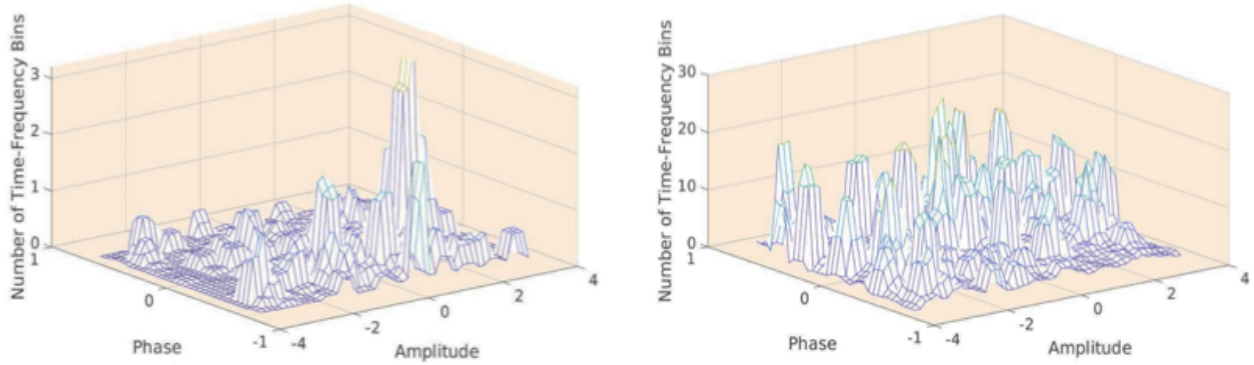


Figure 19: DUET produced histogram predictions of the delay of the true sources calculated from the number of time frequency bins of each source (right delay estimation of pure c-tone, left delay estimation of acoustic reflection of speech).

Figure 20: Photo of beamforming simulation using omnidirectional Blueball microphones left. Photo of prototyped system right.