Sauron Surveillance System Midyear Design Review Report

Jose LaSalle, Omid Meh, Walter Brown, Zachary Goodman

***Abstract*—Sauron is a security system that can be deployed in crowded areas to eavesdrop on individuals of interest. Sauron is an acoustic beamformer with a camera so that the operator can visually select targets. The beamformer is composed of a microphone array that records sound at different points. When the operator clicks on a target in the video, Sauron calculates the angle to the target and uses enhanced delay sum beamforming to extract what the target is saying.**

***Index Terms*—Acoustic, Source Isolation, Microphone Array, Delay Sum Beamforming, Compound Array.**

I. Introduction

SECURITY is a significant concern in public places, resulting in an increased interest in surveillance. Crowded places such as museums, markets, and airports are swarming with cameras. Sauron is a tool to further improve safety. Sauron allows security personnel to eavesdrop on individuals through the power of acoustic beamforming by simply identifying them in a video feed.

Sauron consists of a microphone array and camera that interface with a computer. An operator will be able to click on an individual in a crowded environment and the system will play what that individual is saying. This system can be adapted to be useful in almost any situation where a voice needs to be isolated. For example, an operator might record a lecture and click on students in the audience who are asking questions. Another use case would be video editing. A cameraman might record something and then want to eliminate a distraction in the background. Although Sauron is meant to improve safety, it has other applications as well.

Sauron is a threat to privacy. If enhanced, it could be deployed in a neighborhood to eavesdrop on conversations inside households and other private locations. The major obstacle in this task would be that potential targets would be at different distances from the array. Closer targets will be louder, meaning that delay sum beamforming would fail unless there were a large number of microphones, all of which would be sensitive enough to hear at a long range.

Sauron consists of a microphone array and camera that interface with a computer. An operator will be able to click on an individual in a crowded environment and the system will play what that individual is saying.

J. LaSalle majors in Electrical Engineering and is a member of Commonwealth Honors College.

O. Meh majors in Electrical Engineering and in Computer Systems Engineering and is a member of Commonwealth Honors Collage.

W. Brown majors in Computer Systems Engineering and in Computer Science and is a member of Commonwealth Honors Collage.

Z. Goodman majors in Electrical Engineering.

A. Established Solution

Squarehead Technology's new AudioScope is a device designed to listen in on players, coaches, and the like at sports events. This device performs acoustic beamforming with an array of around 300 microphones mounted on a disk on the ceiling to isolate locations selected by the operator [2].

Currently, airports have some of the most advanced surveillance. Video feeds are analyzed to identify individuals on watch lists, bags being left behind by their owners, people going the wrong way through checkpoints, and cars spending an abnormal amount of time in the parking lots [1]. However; audio is not as prevalent in airport security.

B. Use Case

A security guard with no knowledge of acoustic beamforming and very little training beyond the norm sits in a video surveillance room. One of the cameras is aimed at a line of people waiting to be screened at a checkpoint. Two individuals with suitcases are chatting near the back of the line. To be on the safe side, the guard clicks on the head of one of the speakers. The conversation can be heard through the guards headphones.

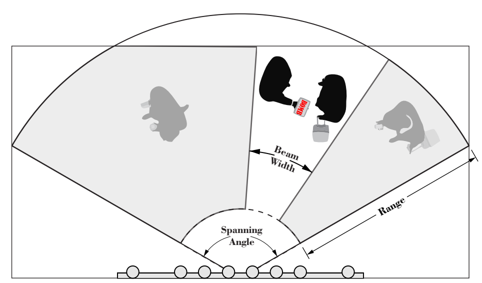


Fig. 1. Visual depiction of specifications.

C. Specifications

Table I lists the specifications of Sauron. The specifications for the targets distance, angle from the arrays center-line, and maximum beamwidth are about the same as the SDP 14 beamforming group had [3]. The SDP 16 group will add context around the array, only increasing the performance of the array where absolutely needed. These old specifications are reasonable for a bottleneck like a corridor.

The beamwidth specification is for a -10dB bandwidth because -10dB is will make a sound seem half as loud to a listener [4]. Tests done within the SDP 16 group showed that

TABLE I
TABLE OF SPECIFICATIONS

Specification	Requirement Value
Range	1 to 3 meters
Angle from center-line to target	-65° to 65°
Maximum -10dB beamwidth	40°
Operating frequency range	1 kHz to 3.5 kHz
Response time	10s
Error in selected angle	20°

when one of two superimposed voices is amplified to 10dB above the other the amplified voice is easy to understand.

Experiments within the SDP 16 group found that higher frequencies were more important for determining what a person is saying than lower frequencies. These experiments involved taking sound clips of group members speaking and running them through a digital bandpass filter, expanding the passband until the message was clear. The specifications were changed to include this useful frequency range, as is reflected in Table I.

As security may need to quickly respond to a conversation, the operator must hear what the target said no longer than 10 seconds after they have said it. Reducing this delay is preferable even over hearing all that the target is saying. When the operator selects on a target, the actual angle that the system is focusing on must be within 20 ° of the intended target. More error than this and the beam will miss the target.

Figure 1 provides a visual depiction of these specifications.

II. DESIGN

Figure 2 shows the layout of Sauron. Sauron will use a fisheye camera, a microphone array, and a computer. The video information is sent to the user interface so the operator can pick a target. The user interface will map the target location to an angle which will be used by the audio processing portion of the program to perform beamforming on the microphone data. This will yield an isolated sound that the user interface can play.

A. Microphone Array

The purpose of the microphone array is to record sound from different locations. It will send this information to the audio processing software described in section II-D.

Our array needs to produce high-quality sound across our desired frequency range with a relatively constant beamwidth.

Beamforming involves processing multiple microphone outputs to create a directional pickup pattern. It is important that the microphone only picks up sound from one direction and attenuates the sound that is off the main axis. Beamforming capabilities are determined by the geometry of the microphone array, the polar pattern of the microphones, and the speed of sound (which could be more accurately determined using a temperature sensor).

Information about the geometry of the microphone array and speed of sound are used to determine the time delays used in the beamforming algorithm. The array geometry also influences what frequencies the array operates at. Smaller microphone spacing is optimal for high frequencies while larger spacing is superior at lower frequencies.

1) Microphones: Microphone (or microphone array) directionality describes the pattern in which the microphones sensitivity changes with respect to changes in the position of the sound source. An omnidirectional pattern is equally sensitive to sound coming from all directions regardless of the orientation of the microphone. A cardioid polar pattern means that there is minimum signal attenuation when the signal arrives from the front of the microphone (0° azimuth), and maximum signal attenuation when the signals arrive from the back of the microphone (180° azimuth), referred to as the null. Figure 3 shows a 2-axis polar plot of the omnidirectional and cardioid microphone responses. This plot looks the same regardless of whether the microphones port is oriented in the x-y, x-z, or y-z plane [5].

The cardioid polar pattern offers beamforming capabilities by creating a beam where the signal is attenuated except for where the beam is steered, while an omnidirectional polar pattern has no attenuation in any direction relative to the microphone. A cardioid polar pattern with a wide angle of operation and narrow beamwidth is desired from our beamforming array in order to focus our beam on a single individual and operate in the largest area possible. We will use back baffled omnidirectional MEMS microphones for our array to create cardioid polar patterns for our operating frequency range. MEMS stands for Micro-Electro-Mechanical Systems, which include microsensors and microactuators that act as transducer elements that convert acoustic pressure waves into electrical signals [6]. MEMS microphones enable improvements in sound quality for multiple-microphone applications. Microphone arrays can take advantage of thee small form factor, sensitivity matching, and frequency response of a MEMS design for beamforming to help isolate a sound in a specific location [7].

High Input sound quality is the result of high sensitivity microphones, a uniform output level across our operating frequency, and low noise.

Microphone sensitivity is defined as the ratio of the analog output voltage to the input pressure. The standard reference input signal for microphone sensitivity measurements is a 1 kHz sine wave at 94 dB sound pressure level (SPL), or 1 pascal (Pa) pressure. Microphone sensitivity is determined using the reference input signal. As microphone sensitivity increases, the output level for a fixed acoustic input increases. Microphone sensitivity measured in decibels (dB) is a negative value, meaning that higher sensitivity is a smaller absolute value [8]. The sensitivity of the microphone array is higher than that of each individual array because their outputs are summed.

- Cardioid
  - -54dBV sensitivity
  - 50-15kHz frequency range
- Electret
  - -44dBV sensitivity
  - 20-20kHz frequency range

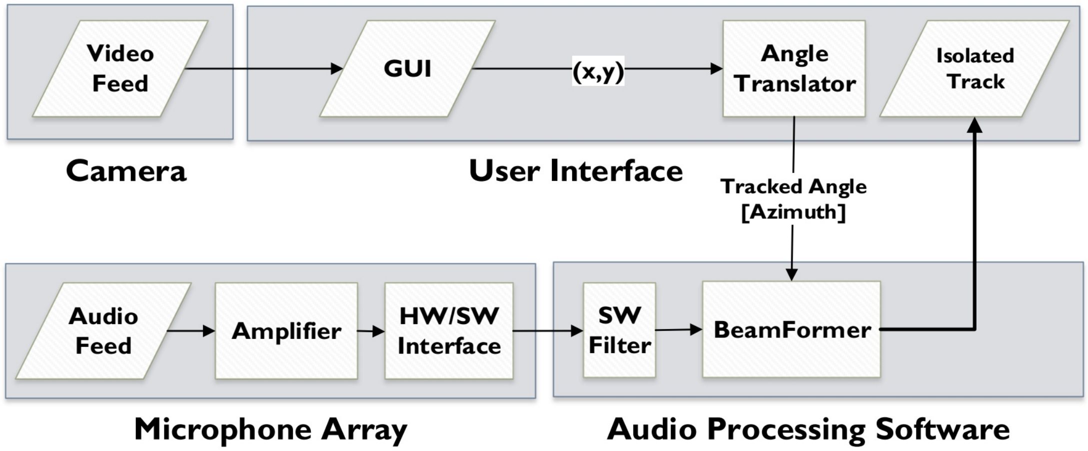


Fig. 2. System diagram for Sauron.

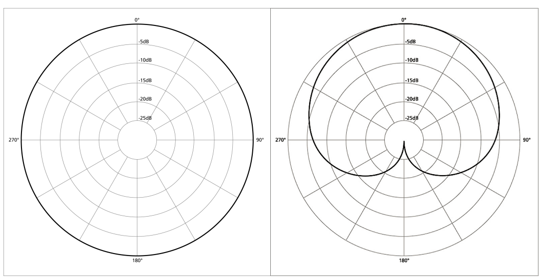


Fig. 3. Free-field polar sensitivity plot of omnidirectional (left) and cardioid (right) microphones.

- MEMS
  - -38dBV sensitivity
  - 100-15kHz frequency range

The frequency response of a microphone describes its output level across the frequency spectrum. The high and low frequency limits are the points at which the microphone response is 3 dB below the reference output level (normalized to 0 dB) at 1 kHz. Figure 4 shows the frequency response of the ADMP510 omnidirectional MEMS microphone [5].

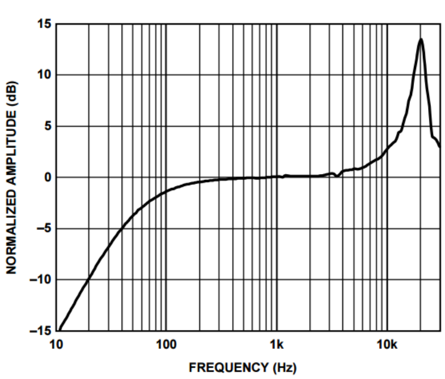


Fig. 4. Frequency response of ADMP510 MEMS microphone.

When building the microphone array, knowing the frequency response of a microphone allows us to design a preamplifier with the appropriate gain to adjust the microphone signal level to match the desired input level of the rest of the circuit or system. Knowing the microphones frequency response enables us to choose microphones based on what frequency range we want to cover. In our desired operating range (1kHz - 3.5kHz), we can see that MEMs microphones have a flat, linear frequency response, meaning we do not have to attenuate or amplify our signals differently at different frequencies to achieve a uniform output across the frequency spectrum.

Low noise is essential for high quality audio. Following the microphones, op amp and difference amplifier circuits are available with significantly lower noise than the microphones themselves, making the microphones the limiting factor in the noise of the overall design. The cable connections must be shielded and/or filtered to prevent the wires from picking up electromagnetic interference (EMI) or RF noise.

By using an array of high sensitivity microphones, low noise preamplifier circuitry, and shielded transmission wires, we can achieve high quality audio input into our computer interface for frequencies based on the array geometry. We are limited in the array design, as more microphones and more powerful ADCs or DSPs exceed our budget.

2) Array Organization: The geometry and the number of elements in the array directly affect the performance. In general, as the target frequency in a linear array is increased, the beamwidth is decreased. To understand this, look at Figure 5. The signal is parallel to the mic array which is gives us the maximum delay. The phases for the 500Hz signal arrive at the microphones in the array at [0 36 73 110] degrees, which are close and difficult to distinguish in terms of coherency. However, as the frequency increases the phases for the 1500Hz signal arrive at [0 110 146 330] degrees. For higher frequencies, the maximum phase difference becomes larger and during the analysis it will be easier to distinguish how incoherent signals due to large phase differences in the received signal.

In other words, the larger phase differences allow us (as long as we are in the same cycle) to determine the direction of the source more clearly, thus giving the microphone array higher directivity. Notice that the directivity is different for different frequencies, as for different frequencies we have different range of arrival phase difference.

Smaller microphone spacing is better for high frequencies and larger microphone spacing is desirable for lower frequencies. To achieve the best result for all frequency bands, we will use a compound microphone array, which is the superposition of multiple arrays with different microphone spacings. Bandpassing the signal to the proper frequency range for each array and subarray, performing the delay-sum for the specific band, and finally, summing the results of the different bands to obtain result with maximum beam precision for multiple frequency bands. Using equal microphone spacing prevents the array to create a precise beam for a wider range of frequencies. Figure 6 depicts the layout for the compound array we plan to implement.

3) Analog to Digital Converter: Measurement Computings USB-204 Data Acquisition Device [9] is the A/D used for this project to handle the microphones. SDP 14 used this A/D and SDP 16 started with their array [3]. This A/D can sample above the needed Nyquist rate of 7 kHz.

This A/D only supports 8 microphones. However, it seemed that A/Ds that supported more inputs would be outside of our price range.

To demonstrate the real-time functionality of the A/D with the rest of the array, a Matlab script was run that took the output of the A/D and echoed it back on computer speakers. A group member yelled into the microphones and could be heard from the computer speakers. In this test, the signal from the array was incredibly noisy. However, this was attributed to the microphones and amplifiers that were used.

The MDR demo also showed the use of a Matlab script that could estimate a clappers angle from the array based on the delays between where the clap was heard between the microphones. This could be useful for calibrating the camera angles.

B. Camera

The purpose of this block is to produce a video that the operator can reference to choose a target to listen to.

This will produce visual data that will be displayed by the user interface described in section II-B.

A fisheye camera will be used to give a wide field of view. Matlab code will need to be written to interface with the camera via USB. Matlab will be able to interface with the webcam [11].

The camera's functionality can be tested by attaching it to a computer and looking at the .avi files produced.

C. User Interface

The purpose of this block is to let the user easily interact with the system.

This will be a graphical user interface that takes video information from the camera described in section II-B and displays it. The user will be able to click on a target in the video to listen to. This block will then calculate the angle from the center-line of the array described in section II-A to the target. This value will be sent as an input to the audio processing software described in section II-D. The audio processing software will calculate and provide the audio coming from the selected point so that the user interface can play it to the user.

The interface will be made in Matlab.

This block can be tested by having a human user observe the system respond to selecting an individual on the video feed and and hearing the audio.

D. Audio Processing Software

The purpose of this block is to isolate the target's voice. It is given the angle to the target by the user interface described in section II-C. It is gets the necessary audio data from the microphone array described in section II-A. This block gives the isolated voice of the target to the user interface.

Audio information is sent through a noise reduction filter as described in section II-D2. This filtered audio is then used for beamforming to the target angle as described in section II-D1. Both will be implemented using Matlab.

The individual components within this block can be tested by inputting a sound file and listening for the expected change. The block as a whole can be checked in this fashion as well.

1) Noise Reduction: Noise is the main limiting factor in designing for a high SNR. This system can be electronic and acoustic. In addition to hardware optimization for electronic noise, we also implement an LMS adaptive filter in our Matlab software to account for acoustic noise. An adaptive filter is a computational device that iteratively models the relationship between the input and output signals of a filter. An adaptive filter self-adjusts the filter coefficients according to an adaptive algorithm. Figure 8 shows the diagram of a typical adaptive filter.

The linear filter can be different filter types such as finite impulse response (FIR) or infinite impulse response (IIR). An adaptive algorithm adjusts the coefficients of the linear filter iteratively to minimize the power of e(n). The LMS algorithm is an adaptive algorithm among others which adjusts the coefficients of FIR filters iteratively [12].

Figure 9 is an image representing noise cancellation. The top graph Noise represents the noise floor of the room. The noise floor measurement necessitates the absence of acoustic signals such as human voice and the presence of typical noisy appliances such as an HVAC. The middle graph Noise + Signal is a visualization of someone speaking in the room before the LMS is applied to output the noise-reduced Signal. This filter works fairly well, and we intend to implement it in our final design.

2) Delay-Sum Beamforming: Beamforming, also known as Spatial Filtering, is a signal processing method used with sensor arrays allowing directional reception or transmission of the signal. For this project we are interested in directional reception of the human voice. Since the speech is a broadband signal, we decided to use a delay-sum beamforming with a linear array which allows us to process a wideband signal and relatively low computational complexity.

Figure 10 is an illustration of a simple microphone array composed of three microphones and a summing module. As

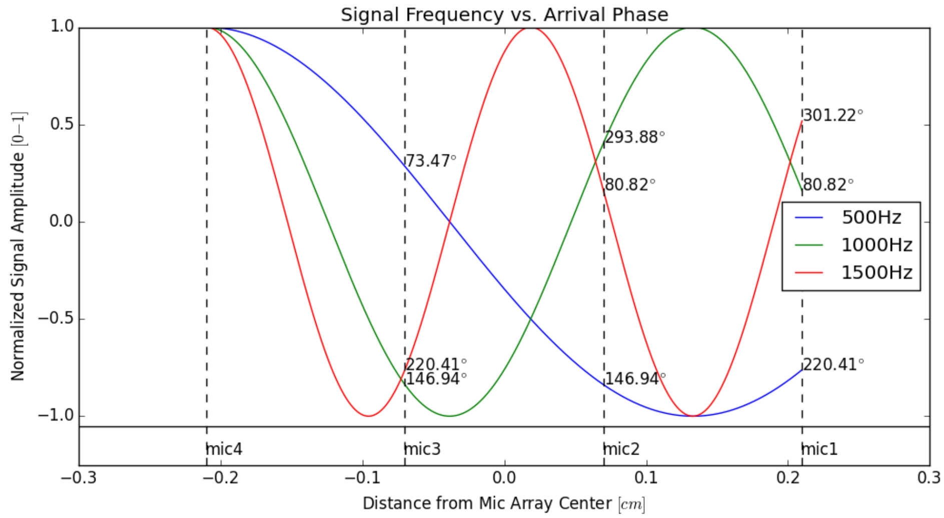


Fig. 5. Wave phases over time for different frequencies.

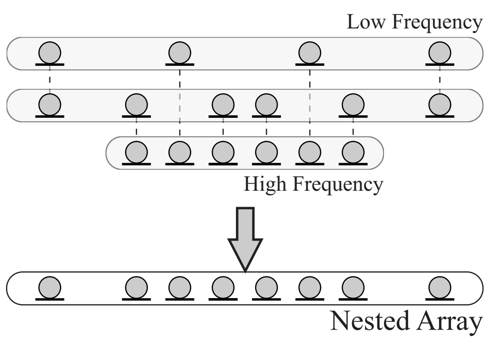


Fig. 6. Drawing of our compound array design. The low frequency array has a spacing of 21cm. The array for middle frequencies has a spacing of 14cm, except for he middle two microphones which are 7cm apart. The highest frequency array has a spacing of 7cm.

shown, when the signal is produced at the -45° it arrives at the left, middle, then right microphones in order, and when the signal is produced at the +45° angle it arrives at the right, middle, then left microphones in order. In both cases, when all three signals are summed the signals will be off by some time delay and will not constructively add up. However, if the signal is produced perpendicular to the array, it arrives at the three microphones at the same time resulting in a constructive signal sum. This microphone array is called a non-steered (focused on 0° azimuth) 3-element linear microphone array.

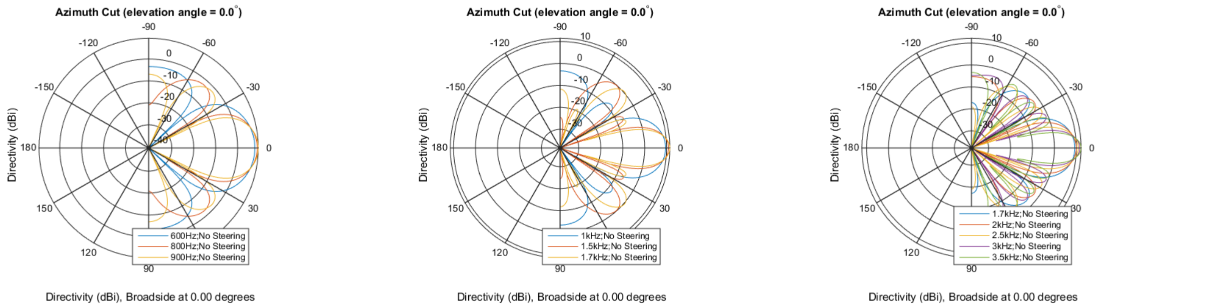
As illustrated in Figure 11, This concept can be further expanded to steer the array beam to an arbitrary direction. A delay block is added to each signal before the summer which further delays the signal. The added delay is to reverse the expected time delays for the signal coming from the desired direction. For instance, in Figure 11, we desire to listen to the target wavefront (top speaker), this we mathematically calculate the expected time delay for the signal to arrive at each microphone. Next, the received signals are shifted back in time (in the steering unit) to look as if they were all received at the same time by mics. At the summing stage, this will result in the constructive interference for the signals coming from the target direction and destructive- or incoherent- interference for the signals coming from other directions.

TABLE II
MDR GOALS

MDR Goal	Status
Voice Isolation Between Two Individuals	Accomplished
Implement SDP14 Array	Accomplished
Implement Self-Made Mic Array	Accomplished
Calibration System to Identify Time Shifts	Accomplished
Beam-Forming via Time Shifting	Accomplished
Noise/Interference Filter Implemented	Accomplished

III. PROJECT MANAGEMENT

Our team has shown a lot of vitality and perseverance since the beginning of this project, and through that we continue to learn how to work together efficiently and effectively. With communication and personal accountability as our mode of operation, coupled with frequent meetings and clearly delegated tasks, we were able to accomplish all of our MDR goals despite a late start. We achieved our goal of demonstrating voice isolation between two speakers by establishing four specific sub-goals that were tailored to each team member's area of strength. Analysis of the hardware for the mic array was headed by Zach, as amplifier design and use of electronic elements are in his field of study as an electrical engineer. Walter was responsible for interfacing the hardware into Matlab and building a software block for calibrating the array. Omid took on the beam-forming algorithm given his CSE background, and Jose was responsible for noise reduction as this was involved in his REU. As an execution of our plans unfolded, an overlap of our knowledge bases lead to a very integrated experience of one helping the other, resulting in a very rewarding experience so far.



(a) Low Frequency Subarray (b) Mid Frequency Subarray (c) High Frequency Subarray

Fig. 7. Simulation results for sub-arrays within the system. 7a is the low frequency array in the Fig 6 with four elements at $21cm$ spacing. 7b is the middle frequency array in the Fig 6 with six elements at $[14cm\ 14cm\ 7cm\ 14cm\ 14cm]$ spacing. 7c is the high frequency array in the Fig 6 with six elements at $7cm$ spacing. 7a is tuned for [600Hz, 1kHz], 7b is tuned for [1kHz, 1.7kHz], and 7b is tuned for [1.7kHz, 3.5kHz]

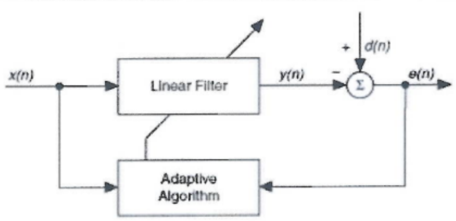


Fig. 8. Diagram of adaptive filter for noise reduction. x(n) is the input signal to a linear filter, in this case the noise floor of a room. y(n) is the corresponding output signal. d(n) is an additional input signal to the adaptive filter, in this case the recorded noisy signal. e(n) is the error signal that denotes the difference between d(n) and y(n), which is the signal.

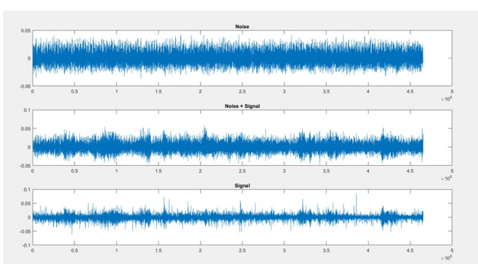


Fig. 9. Demonstration of noise reduction. The top graph shows the noise that is used to calibrate the noise reduction algorithm. The middle graph shows the raw microphone input, with both the signal and the noise. The bottom graph shows the result after the noise reduction is applied.

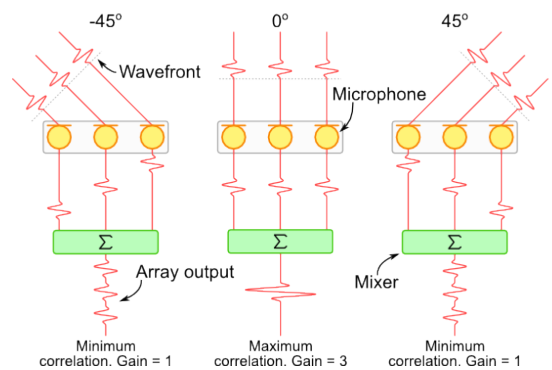


Fig. 10. Simple microphone array with sounds coming from the direction of -45°, 0° , and 45°. Reprinted with permission from [13].

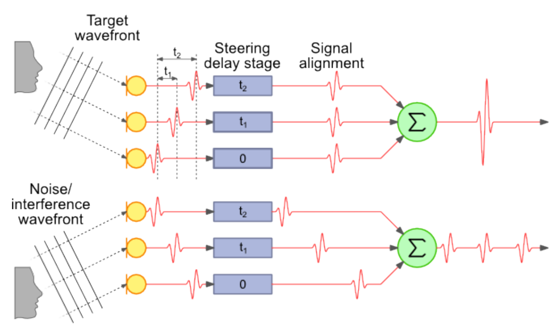


Fig. 11. Illustration of delay-sum beamforming. Reprinted with permission from [13].

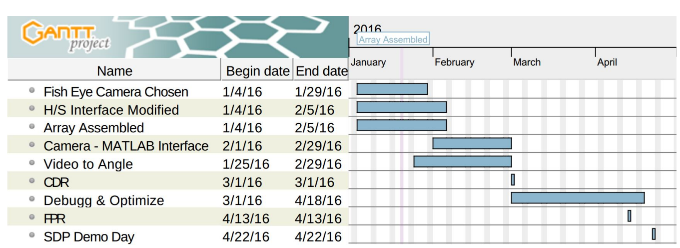


Fig. 12. Gantt chart of the schedule for the development of Sauron.

IV. CONCLUSION

Project Sauron is proceeding on schedule. Table III details our MDR deliverables and shows they were completed on time. These deliverables demonstrated that the group could interface with an array and that the group could isolate voices. MDR has resolved the physical boundaries that the group feared would stop them.

For CDR, the group intends to demonstrate that a user can click on a point in a fisheye video feed and Sauron will isolate the audio at that point. Figure 12 shows what milestones will need to happen along the way. A new array must be built to support the tight beamwidth called for by the specifications. A fisheye Camera must be acquired so that there will be time to interface with it. Ideally, there will be a good mapping between the video and the target angle by this point. Then integration will be the last step needed.

The major challenge will be to properly map the coordinates in the video feed to match the angle of an object. This system needs to be incredibly accurate due to the thinness of the beam. Although this process can be started before the fisheye is acquired, it will be impossible to complete it with certainty until the interface with the fisheye is working.

Assuming all goals are met for CDR, the group hopes to experiment with extra features that will improve the usability of the system, such as the ability to track moving targets and improving Saurons reaction time.

A. Acknowledgments

We would like to thank Professor Hollot and Professor Moritz for their feedback and guidance in establishing realistic goals. We would also like to send a big thanks to Professor Wolf who took the time to meet with us each week and helped us stay on track and organized. An additional thanks for Alumni John Shattuck for coming back to UMass to meet with us as we evolve his old project.

REFERENCES

[1] Airports [Online]. Available: https://www.videosurveillance.com/airports.asp [Accessed Web. 18 Jan. 2016.]
[2] Catherine de Lange Audio zoom picks out lone voice in the crowd [Online]. Available: https://www.newscientist.com/article/dn19541-audio-zoom-picks-out-lone-voice-in-the-crowd/ [Accessed Web. 21 Jan. 2016.]
[3] J. A. Danis, et al. *The Acoustic Beamformer*. Available: http://www.ecs.umass.edu/ece/sdp/sdp14/team15/assets/Team15Final MDRReport.pdf [Accessed Web. 18 Jan. 2016]
[4] University of WisconsinMadison, About Decibels (dB) [Online]. Available: http://trace.wisc.edu/docs/2004-About-dB/ [Accessed Web. 24 Jan. 2016.]
[5] InvenSense, Microphone Specifications Explained [Online]. Available: http://43zrtwysvxb2gf29r5o0athu.wpengine.netdna-cdn.com/wp-content/uploads/2015/02/MICROPHONE-SPECIFICATIONS-EXPLAINED.pdf [Accessed Web. 2 Dec. 2015.]
[6] InvenSense, Analog and Digital MEMS Microphone Design Considerations [Online]. Available: http://43zrtwysvxb2gf29r5o0athu.wpengine.netdna-cdn.com/wp-content/uploads/2015/02/Analog-and-Digital-MEMS-Microphone-Design-Considerations.pdf [Accessed Web. 2 Dec. 2015.]
[7] Digi-Key, MEMS Technology for Microphones in Audio Applications [Online]. Available: http://www.digikey.com/en/articles/techzone/2012/aug/mems-technology-for-microphones-in-audio-applications [Accessed Web. 2 Dec. 2015.]
[8] Analog Devices, Understanding Microphone Sensitivity [Online]. Available: http://www.analog.com/library/analogDialogue/archives/46-05/understanding_microphone_sensitivity.pdf [Accessed Web. 2 Dec. 2015.]
[9] Measurement Computing, USB 200 Series12 bit DAQ Device With 8 Analog Inputs [Online]. Available: http://www.mccdaq.com/usb-data-acquisition/USB-204.aspx [Accessed Web. 1 Nov. 2013.]
[10] Mathworks, Database Toolbox [Online]. Available: http://www.mathworks.com/products/database/ [Accessed Web. 23 Jan. 2016.]
[11] Mathworks, Acquire Images from Webcams [Online]. Available: http://www.mathworks.com/help/supportpkg/usbwebcams/ug/acquire-images-from-webcams.html [Accessed Web. 24 Jan. 2016.]
[12] National Instruments, Least Mean Square (LMS) Adaptive Filter [Online]. Available: http://www.ni.com/example/31220/en/ [Accessed Web. 18 Jan. 2016.]
[13] A. Greensted, Delay Sum Beamforming, The lab book Pages An online collection of electronics information, 01-Oct-2012. [Online]. Available: http://www.labbookpages.co.uk/audio/beamforming/delaySum.html. [Accessed: 24-Jan-2016].